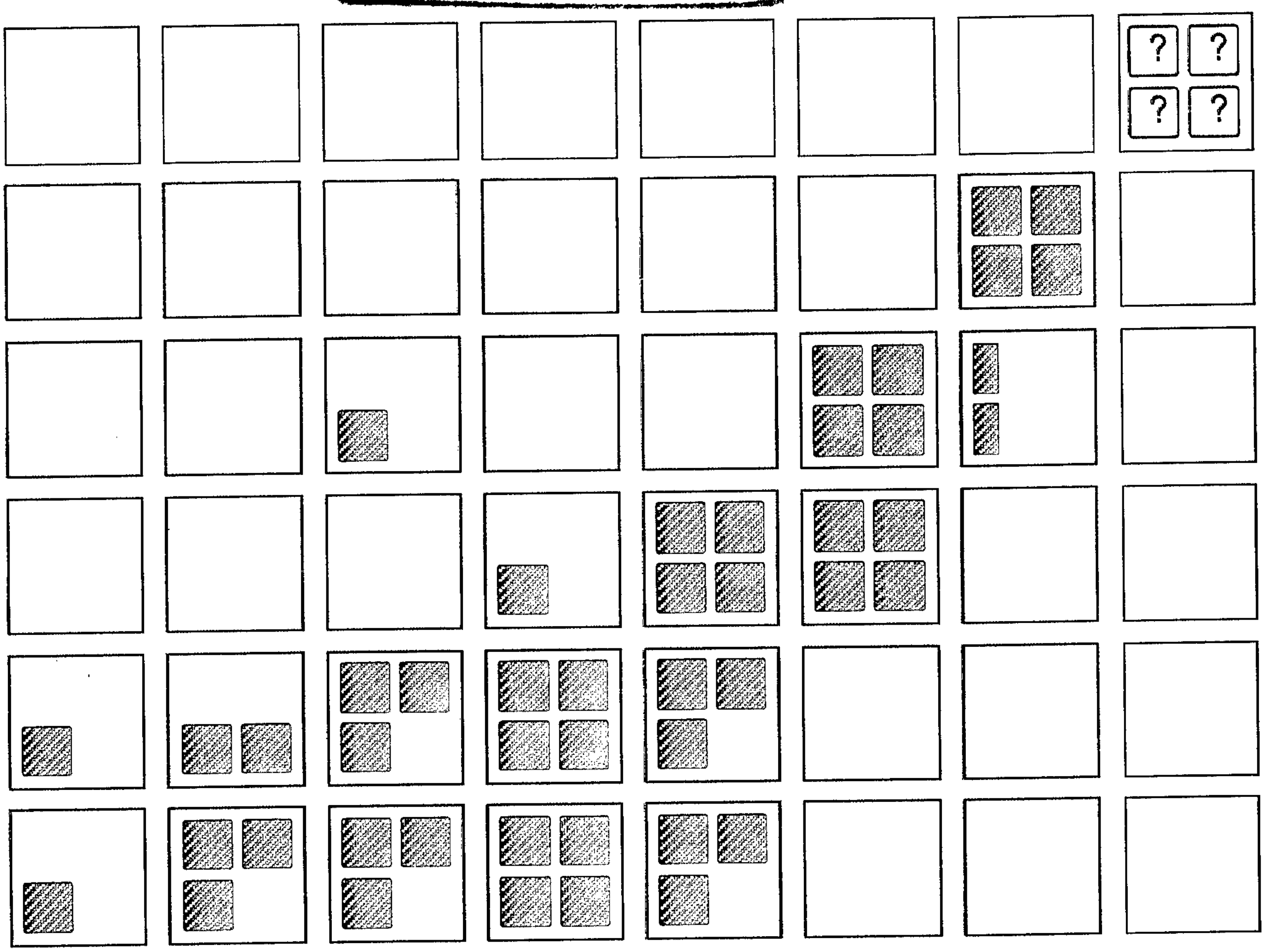
TOWARD AN INTEGRATED ENVIRONMENT FOR WARFIGHTING CONTROL

DISTRIBUTION STATEMENT A
Approved for public release;
Distribution Unlimited



NAVAL SEA SYSTEMS COMMAND
NSWC
Naval Surface Warfare Center
DAHLGREN DIVISION
Panama City Dahlgren

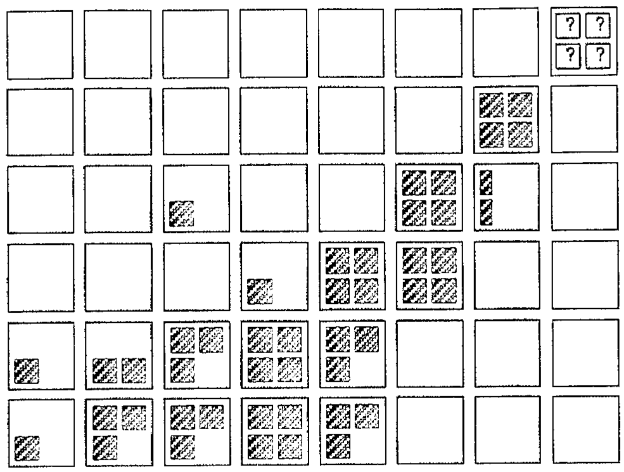
COMBAT SYSTEMS DEPARTMENT
SEPTEMBER 1997

DTIC QUALITY INSPECTED 4

19971117 068

NSWCDD/MP-97/54
Approved for public release;
distribution is unlimited.

TOWARD AN INTEGRATED ENVIRONMENT FOR WARFIGHTING CONTROL



Page

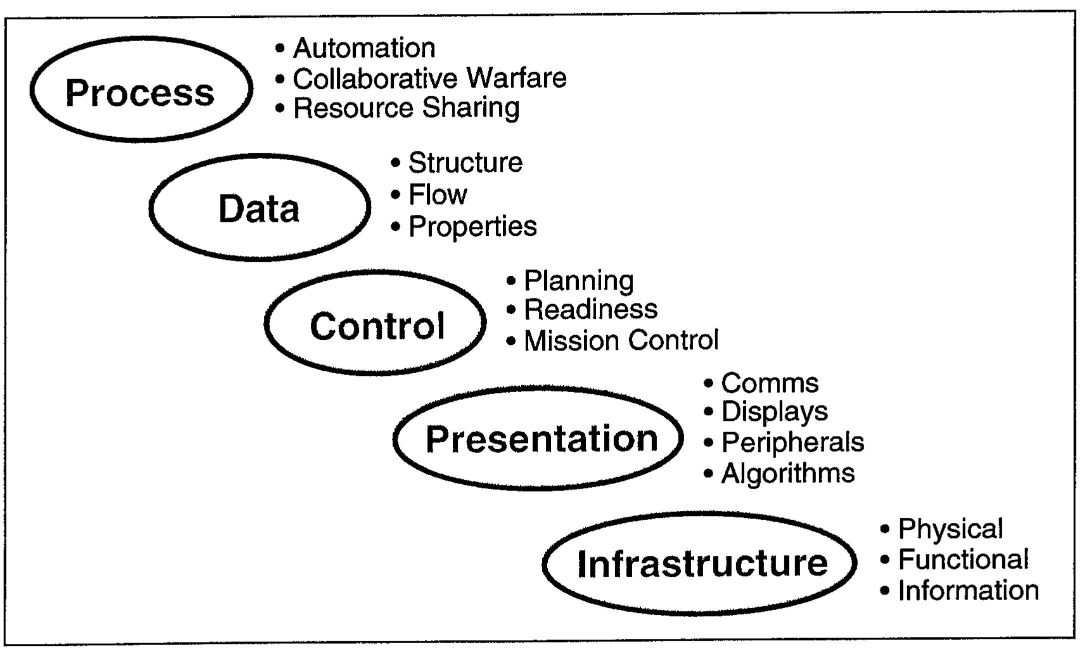
SEPTEMBER 1997

COMBAT SYSTEMS DEPARTMENT
NAVAL SUFACE WARFARE CENTER DAHLGREN DIVISION
DAHLGREN, VIRGINIA 22448-5100

EXECUTIVE SUMMARY

This report is based on the premise that total ship system engineering should seek to maximize value delivered to mission teams on a life cycle basis. The first step in defining a ship is to consider its purpose and how warfighters envision its use. Primary emphasis is on understanding what future mission teams must do, what the associated operating processes will be like, and what information will be needed to execute those processes. Success is achieved only when an executable system design description has been produced and acceptance criteria for the ship have been nailed down. These criteria reflect how the user will see the ship and key mission systems (and interact with them) as the tasks laid out by the concept of operations are performed.

In particular, the report considers what characteristics make a fighting unit such as a ship or warfare system into a system of systems. A system of systems is one formed by integrating other systems, each a product in its own right with its own development team, objectives, management, and schedule. The core problem is not how to break a ship into component systems but how to articulate what it means for component systems to work together as a single entity. Emerging strategies for the engineering of complex systems, such as naval ships and warfare systems, lead to the flowdown of integration objectives shown below. Suitable guidelines might be created, for example, by forming a cross-functional team for each of the integration objectives shown.



FLOWDOWN OF INTEGRATION OBJECTIVES

INTRODUCTION

This report, along with References 1 through 3, results from an effort to reinvent the Navy's process for transforming operational needs into warfare systems and combatants. The overall objective is to articulate a framework and strategy that will foster teamwork across organizational lines in creating a new generation of warships, designed from the keel up to enable on-board mission teams to act as integral parts of a joint operating force and to set a new standard in life cycle effectiveness. This report expands on the backbone control concept featured in the earlier reports.

Results of the previous work suggest that a common environment for control of operating processes (control backbone) can be created, based on systematic integration of automated control and support resources across mission teams and tasks. Key capability objectives for the envisioned approach are given below.

- Command spaces will become utilities, tailorable to any required set of mission teams, organic or embarked. Any console, at any location, will support any operator position.

- Command spaces will be reconfigurable and flexible, so that watch station functionality and layout can be tailored to any mission or task required. Changes in technology and operating process will be easy to accommodate. Software will aid a Commanding Officer in tailoring ship organization and arrangement to assigned mission tasks and personnel.

- Incompatibilities in control features of different mission areas will be eliminated without requiring all man-machine interfaces to be uniform. The aim will be to offer standardized capabilities that can be tailored to the needs and skills of different tasks and operators.

- Information assets will be managed on a shipwide basis, using common C4I interfaces across a diverse mix of installed systems, including some with different age and logistics support characteristics.

- Readiness management and resource control functions will be handled on a shipwide basis. Future ships will track physical plant configuration and utilize current performance data to construct a feedback loop from operations to posture (and to design).

- A scalable open architecture will enable new capabilities to be installed while legacy systems are retained and excessive recertification effort is avoided. The basic design will be reusable across many ship types, serving as the nucleus of a more advanced system with hooks installed to enable change. Extensive use will be made of commercial components to meet affordability goals and to create technology insertion opportunities.

The potential benefits appear very significant. In concept, each mission team will come on board carrying any mission-unique computer programs and data needed, configure watch stations and displays, establish sensor networks to support operations, install the preferred mix of weapons, conduct test and training exercises, and commence operations. Everything but mission-unique assets will be provided by the backbone, including watch stations, interfaces, displays, computers, and networks. Command spaces will be designed for multipurpose use, with layout and functionality tailorable to any necessary mission or task. In advanced systems, it is envisioned that command spaces and warfighting control systems will be designed to support networks of mission teams and component equipments extending across an entire theater of war. Open systems and design for reuse will make changes and upgrades faster and easier.

DOMAIN OF INTEREST

A warship is a complex of people, plant, and procedures that together form a warfighting system of systems. Figure 1 identifies key mission areas and development trends within the domain. Mission control systems create value by mediating the execution of operating processes under the direction of mission teams. This report considers concepts for integration of mission control resources on a total ship basis.

A number of issues and opportunities in the design of mission control facilities can be identified. One has to do with the basic organization of mission resources. U.S. military strategy places increasing

Theater Air Warfare	*◦ Airspace Control in MOOTW* *◦ Theater Missile Defenses* *◦ Automated Force ID/TEWA*
Maritime Warfare	*◦ Integrated Survivability* *◦ Detect and Neutralize Mines* *◦ Shallow Water ASW*
Land Attack Warfare	*◦ NSFS, ASu, STK Integration* *◦ Embarked Vehicles & UAVs* *◦ Automated Strike Control*
Command, Control & Information Warfare	*◦ Unified Tactical Picture* *◦ Auto Plan Execution Options* *◦ Collaborative Operations*
Embarked Elements	*◦ Armed Helicopters* *◦ Unmanned Vehicles* *◦ Special Mission Modules* *◦ Marine Corps/NSW Teams*

FIGURE 1. WARFIGHTING CONTROL TRENDS

emphasis on theater warfare conducted by joint and combined operating forces. As suggested by Figure 2, major warfare systems are thus becoming network systems with nodes at sea, in the air, and on land. This will alter how naval forces and ships are organized to deal with their external partners, competitors, and stakeholders. The mission teams carried by future surface combatants will fight as part of a theater organization that integrates sea, air, land, and space elements to deliver timely and effective theater warfighting capability. The network must reflect applicable doctrine and operating concepts, mission architecture, decision making procedures, and course of action alternatives at unit, force component, and theater levels of concern.

Given this context, mission control requirements are driven by a complex set of trade-offs. While capabilities provided to mission teams depend largely on decisions made in major warfare system development programs, there is potential for affordability and capability gains from integration across warfare systems and ship classes. Increasingly, resources used by one warfare system may be furnished by equipment from another warfare system program or surface ship program.

Integration strategies must also reflect emerging concepts for dealing with people and teamwork, information, processes, and infrastructure.

DIMENSIONS OF INTEGRATION

There are many ways in which any given set of components (weapons, sensors, watch stations, applications, and mission teams) can be assembled to produce an environment that supports mission control. The key question is, How can the right set of components be produced and integrated? Dealing with this question means reexamining the meaning of integration itself. A discussion of the characteristics desired in an integrated environment for mission control is given below (see also References 4 and 5). Figure 3 lists the topics considered. The first topic, process integration, deals with integration across teams and tasks. Intrateam aspects are addressed in the remaining sections of this report. This provides a model of the problem that is general enough to permit consideration of backbone components, their interconnections, and the circumstances in which they will be used.

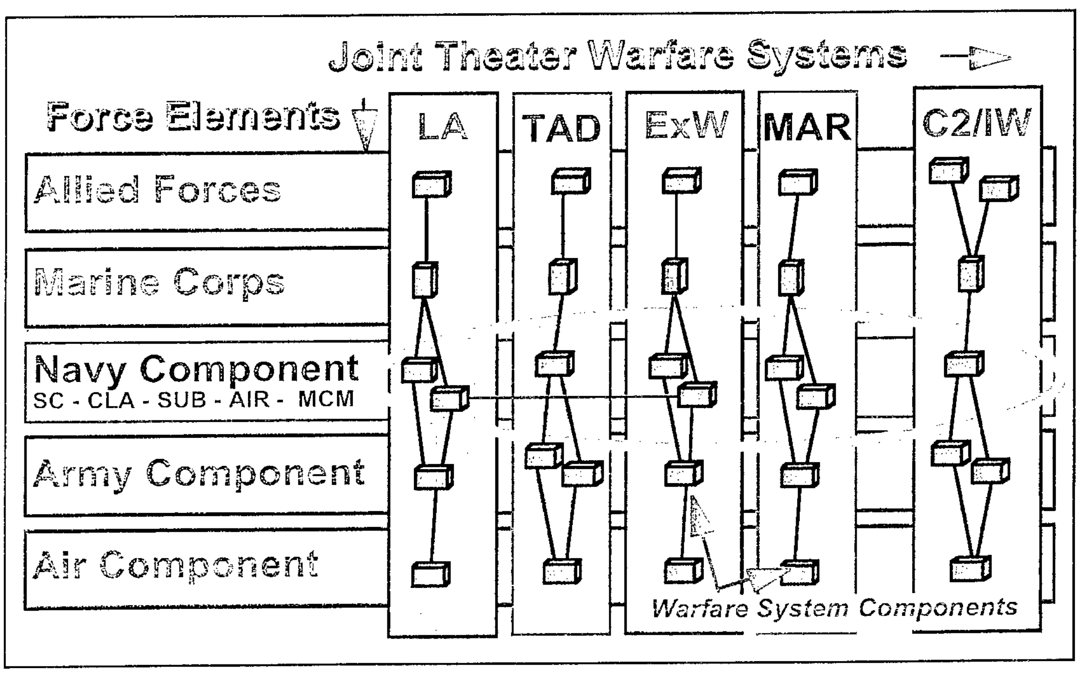


FIGURE 2. FORCE VS. WARFARE SYSTEM INTEGRATION MATRIX

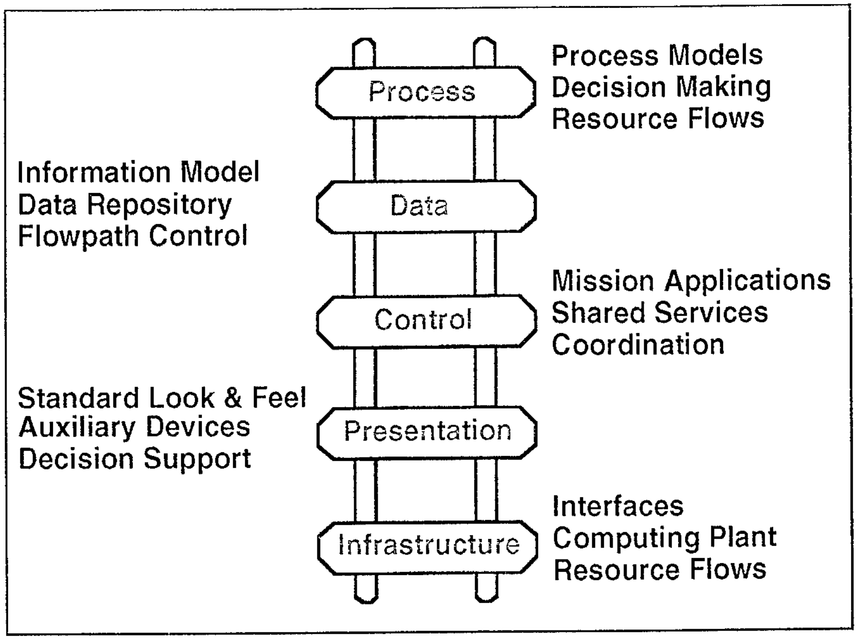


FIGURE 3. INTEGRATION DIMENSIONS

- **Process Integration.** The goal of process integration is to ensure that teams and tasks interact efficiently in support of a defined operating process. Mission teams and operations are generally well integrated when goals and constraints established for each mission team or task, and the implementing sequences of operational activities, are consistent.

- **Data Integration.** The time lines and information flows required to implement a concept of operations are the focus at this level. The design for integration describes how information will be used to support decision making and identifies data elements and data structures that capture the information in the operating process. A basic goal in this area is to provide for a common tactical picture with the ability to maintain several different and specialized views and to propagate changes between such views.

- **Control Integration.** In this area the goal is to enable flexible use of people and resources to accomplish mission tasks in accordance with user preferences. Mission teams must share functional resources to support flexibility in team structure and access to resources. Control integration, in this regard, is complementary to data integration. The latter deals with data entry, storage, management, and access issues, while the former deals with control transfer and service sharing.

- **Presentation Integration.** The goal here is to improve the efficiency and effectiveness of the user's interaction with the environment by reducing his cognitive load. Display management functions are particularly important. Key properties are appearance and behavior integration and interaction paradigm integration.

- **Infrastructure Integration.** At this level we are concerned with commonalities between the control backbone and the shipwide computing environment. Key concerns include interface standards, interconnects, and shared use of resources.

Use of the term "dimensions of integration" is not intended to imply that the factors involved are in some way orthogonal. The message is simply that an integrated environment for mission control can be achieved only if systematic attention is given to these factors and their interactions.

PROCESS INTEGRATION

There are two ways to look at system engineering. One sees the result as a product system, while the other sees it as an operational process. The difference lies in the recognition that a warfighting system's real value is not determined by market forces but by how it is employed, in combination with other systems, to accomplish military objectives in a complex and dynamic operational environment. If mission needs are best defined in an operational context, then more emphasis must be placed on how mission teams operate, how operations are coordinated across teams, and how technology can influence their operations. A clear focus on process concerns is the remedy for stovepiping in development.

The goal of process integration is to ensure that a variety of mission teams and tasks (see Figure 4) interact efficiently in support of a defined warfighting process. Mission teams and operations are generally well integrated when goals and constraints established for each mission team or task, and the implementing sequences of operational activities, are consistent. The benefits expected from systematic attention to warfighting process integration include

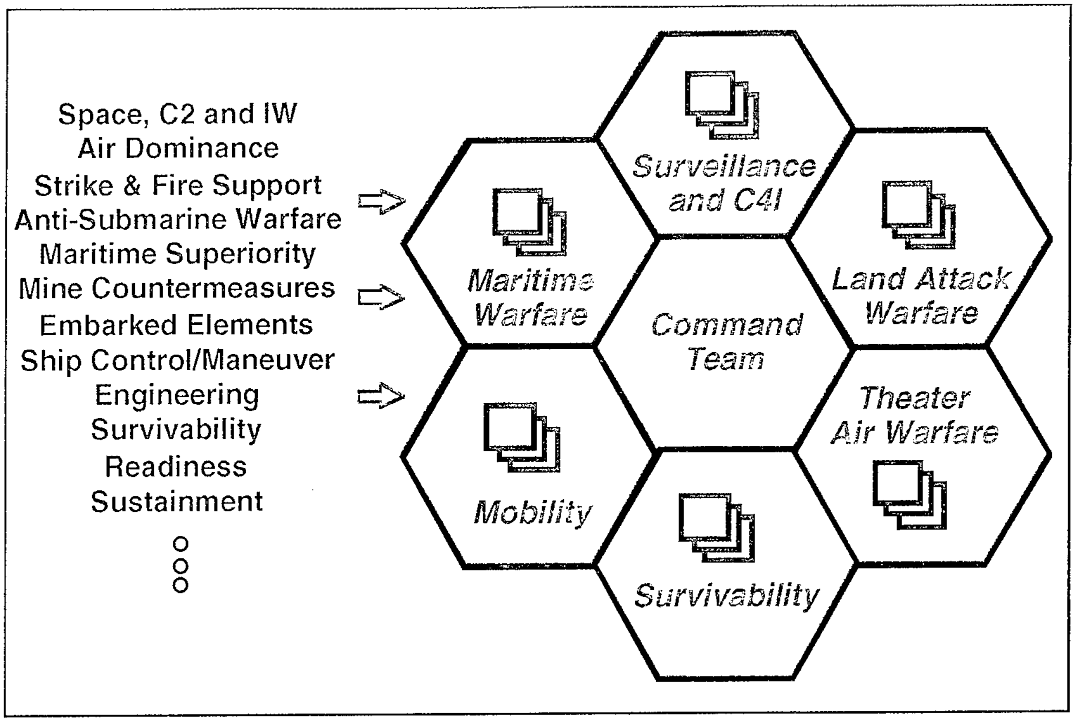


FIGURE 4. ARRAY OF MISSION TEAMS

(1) effective ways to automate without loss of readiness or flexibility, (2) collaborative mission control capabilities, and (3) enhanced life cycle effectiveness due to synergy in development and operation of mission systems. Opportunities for process integration are considered further in the remainder of this section.

AUTOMATION

What the Navy appears to want from automation initiatives is to reduce life cycle cost for a given level of mission performance. Similar efforts have been made by industry in the area of computer-integrated manufacturing (CIM). But CIM has a downside: many times, the resulting systems show a lack of flexibility. As it turns out, people drive flexibility; and when technology issues get more attention than people, systems tend to become inflexible. Since flexibility is an important concern in naval operations, we want systems that combine automation and flexibility. What is needed is automation that will help mission teams in tailoring processes for a real-world operating environment. What is needed is automation that augments human performance in dynamic situations, where the indeterminacy of the world itself ensures that some capability for on-the-fly problem solving will be necessary.

In an era of joint programs, it seems appropriate to take notice of an Air Force automation initiative. The Controls Automation and Task Allocation (CATA) Program described by Reference 6 deals with an integrated control environment for warfighting systems. In Operation Desert Storm, mission packages used by the USAF for precision attack operations generally involved two aircraft, cooperating mainly through preplanned flight parameters and limited voice communication. Today's plan is to adopt a "composite wing" structure, combining a mix of attack and support aircraft to achieve maximum flexibility and strike effectiveness. The role of CATA is to provide an integrated flight management system for the composite wing while limiting the impact on pilot workload. Its functionality will include flight path control, allocation of assets (weapons and sensors), maximizing survivability against threats, limiting fratricide (deconfliction), optimizing fuel reserves, and coordinating attack maneuvers.

In general, it can be expected that future mission control systems will incorporate similar advances in automation. There are opportunities for Navy application in such areas as readiness management, embedded training, damage control, ship control, and ammunition handling. However, better operator interfaces may offer the greatest leverage. If a ship is a weapon system, operator interfaces are the "handle" by which its capabilities are used to conduct warfighting tasks. While improvements may be difficult to quantify, it is clear that new design tools and emerging technologies will permit dramatic improvements in capability and affordability.

COLLABORATIVE WARFARE

U.S. military strategy places increasing emphasis on the conduct of theater warfare by joint and coalition operating forces. Theater warfare involves a different kind of battle space, one in which events

ashore have a great deal to do with mission goals and tasks. Maritime forces add depth to the battlespace and create new opportunities for maneuver. But this isn't a Navy battlespace alone; increasingly, forces ashore and afloat must be able to operate as part of an integrated warfighting system, a network system with nodes at sea, in the air, and on the land.

The Navy's Cooperative Engagement Capability can be cited as an example. In broad terms, cooperative engagement involves the use of distributed sensor and weapon assets to engage air threats. A forcewide infrastructure is provided to control the distributed functionality of the warfighting process. Thus units are interconnected by a covert, jamproof, high-capacity link. The potential for high-altitude and hypervelocity weapons (including tactical ballistic missiles) makes this capability increasingly important.

A joint approach to doctrine, planning, operations, and support will enable these networks to operate as a distributed warfighting enterprise. Command spaces will provide a general task environment for warfighting control, with functionality and layout tailorable to any necessary mission or task. This backbone environment will accommodate any mission team, providing operator interfaces and information flows as necessary to permit available resources to be utilized effectively. In addition, it must help to reduce time and cost to get new technologies into the Fleet. These are key elements of a strategy for making a ship into a real "system of systems."

Many aspects of force integration have their roots in doctrine. One complicating factor in coordination of multiple mission teams is that teams controlling different parts of a joint or coalition warfighting process may employ inconsistent operating concepts. The lack of a unified concept is embedded in the planning, training, decision making, and operating practices of all theater force elements. This is a process integration problem that cuts across at least three levels of doctrine (campaign, task force, and unit). For example, there are important philosophical differences between services on how to approach theater air warfare battle management. This holds especially for differences between Air Force and Navy approaches. For sea, air, and land forces to fight as one, common doctrine will be necessary for mission operations and support functions such as training, connectivity, data standards, decision aids, and logistics.

SYNERGY

Creation of a common environment for mission control means a common core of functional elements, combinable and tailorable to the specific needs of various ship classes. Figure 5 offers a generic view of mission control functionality. The idea is to minimize the proliferation of distinct programs, address pressing interoperability concerns, and gain the life cycle advantage of a single common core to fight, train, and maintain.

Some dramatic changes from traditional designs can also occur in the layout of mission control facilities. Spatial arrangement alternatives include a conventional amphitheater layout, a theater-in-the-

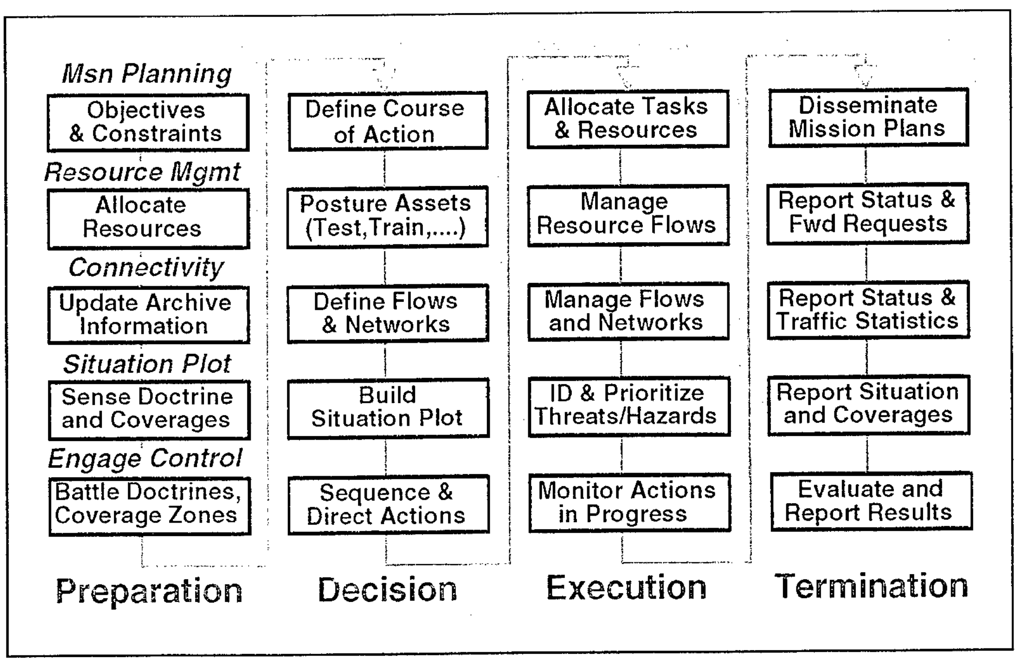


FIGURE 5. MISSION CONTROL FUNCTIONS

round layout, and a distributed layout. Each of the alternatives has certain advantages, and a comprehensive analysis will be needed to determine which would best serve future ships and crews.

In some areas, better capability can be gained by integration across related mission areas. The use of guns, missiles, and air assets for land attack is an example. While the target sets for strike, interdiction, defense suppression, and fire support are somewhat different, the necessary information flows overlap to a great extent. It is possible that a unified fire control system, providing situation plots tailored appropriately for each mission and weapons mix, would be more capable and more affordable than a battery of stovepiped systems. Associated requirements for air space control, monitoring the tactical situation ashore, and adaptability to evolving command and control structures can also be supported by an integrated control environment. Given today's emphasis on joint operations and the promise of sea-based weapons for land attack, an effort to rationalize information and control flows across the entire mission area appears warranted. Figure 6 shows basic functional flows for a consolidated warfighting process. Tie lines drawn across the top of the figure are used to show internal and external information flows. A third tie line, drawn across the bottom of the figure, indicates resource flows.

An integrated control environment also has appeal in the area of survivability. Future joint and expeditionary warfare operations are likely to involve complex operating environments and multiwarfare threats. Dealing with a mix of air, surface, and underwater threats will call for a total ship approach to survivability. This demands an effort that cuts across the traditional boundaries that separate combat and ship systems. Key concepts include automation and integration of signature management, hard kill, soft kill, and passive defenses against threat weapon systems.

DATA INTEGRATION

The goal of data integration is to ensure that all the information in the environment is maintained as a consistent whole, regardless of how parts of it are operated on and transformed to support decision making. This section reviews the character of information flows in warfighting processes and the properties necessary for data integration. Important properties include interoperability, nonredundancy, consistency, exchange, and synchronization.

Until World War II, the pace of operations was such that mission teams could gather selected information, study it, generate alternatives, and evaluate them to determine the most appropriate course of action. The speed of modern vehicles, weapons, and communications stresses the team's ability to react fast enough to a shifting tactical situation. In future conflicts, warfighters are expected to mass fire and not forces. This style of warfighting will make new demands on information management for mission teams. Given the complexity and time constraints of future combat environments, mission teams may have difficulty in responding adequately to the challenges listed below:

- Assign targets with the speed and precision necessary to seize the initiative
- Analyze the tactical situation as quickly and accurately as needed for effective planning
- Generate, select, and integrate responses fast enough to optimize force deployment
- Communicate plans and objectives quickly or accurately enough to maintain unity of effort
- Allocate dispersed resources with sufficient accuracy and speed to position them properly for countering an enemy attack in depth

Consequently, improved data integration is necessary to enhance the efficiency of warfighting processes and continue to reduce manning requirements, while improving time lines for the conduct of operations.

TACTICAL INFORMATION FLOWS

In peace and in war, shipboard mission teams are constantly surrounded by masses of information. Most of it is needed for day-to-day operations, while a small amount makes the difference between success and failure. The challenge is to separate the

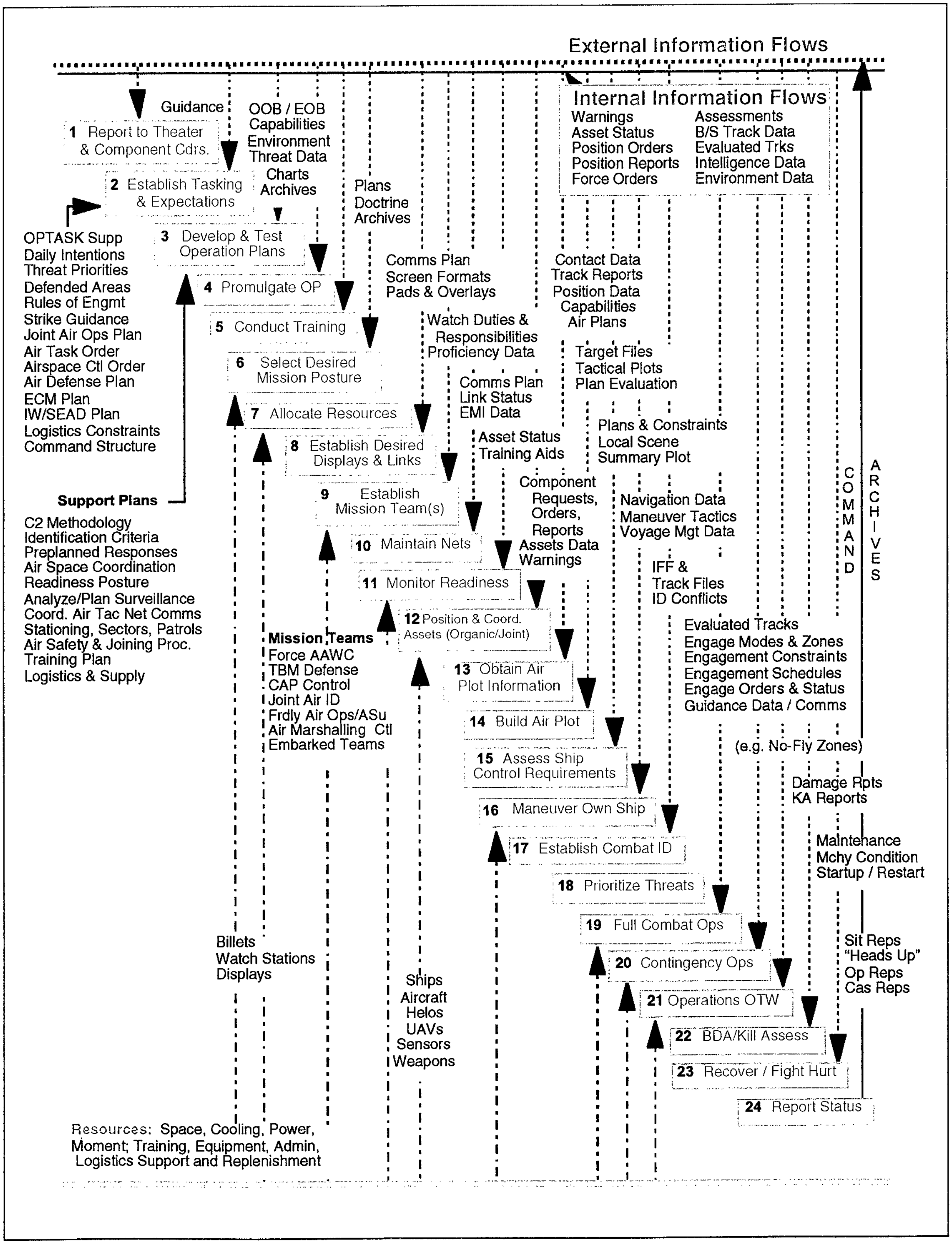


FIGURE 6. PROCESS REFERENCE MODEL - AIR DOMINANCE

wheat from the chaff while remaining flexible enough to handle changing demands. To characterize essential information flows, some key questions must be addressed:

- *What kinds of information must be provided?*
- *What processing modes must be supported?*
- *In what directions must information flow?*

Kinds of Information

A ship must accommodate information relating to all courses of action of interest to decision makers. Mission effectiveness depends on the quality of the tactical picture made available. Ideally, a good situation assessment provides for timely and complete understanding of the military situation at hand and how it has evolved. This includes information on the mission, operating area, enemy situation, own force situation, and weather, along with all essential operational capabilities. The depth of understanding achieved must be enough to support accurate prediction of hostile intent and selection of an optimal course of action. This represents an ideal, an upper bound as to what can be known with confidence.

There appear to be four broad categories of information useful in describing any military situation: identity, geolocation, operational, and organizational. With respect to dynamic behavior, each category contains two kinds of information. One (tactical) is dynamic and perishable, derived from message traffic and organic sensors. This information is most useful for describing the dynamics of evolving tactical situations. The other (archives) consists of fairly static databases, including order of battle data and other nonperishable information that can be used to enhance or refine the overall picture. At a minimum, enough information is needed to answer the questions listed below.

Where am I? (Situation)
Where is the enemy?
What is the enemy doing now?
Where is the enemy vulnerable?
What are the enemy capabilities?
What are the enemy's key decisions?
How should we influence those decisions?
What combat power do we have?
What are our vulnerabilities?
Am I in balance?
Movements? Reserves? Logistics?
How long will it take me to .... ?
How long will it take an enemy to .... ?
What is the most important thing to do now?
How do I get it done?

This list is sometimes called the "commander's catechism." References 7 and 8 provide further discussion. Despite the organizational, procedural, and doctrinal changes taking place in U.S. warfare concepts, these questions remain valid, and serve as a starting point for consideration.

Processing Modes

As a beginning, three modes are used to process the mass of information. Pipeline information is standard, persistent information shared between units. Only a small amount of this information is essential for the commander to get a sense of the battle space. Reports are automatically updated and disseminated per standing operating procedures. Alarms, the second mode, are time-sensitive reports alerting a mission team that things are not going as envisioned and corrective action is needed. This type of information flow is fundamentally important for response initiation in operating process control. Reporting criteria are set either by the commander or by subordinates with an understanding of the commander's intentions and mission objectives. Alarms should not be delayed for any reason and should skip echelons in transmission. Thirdly, the tree mode involves a directed search for information from internal and external sources. The tree represents the many sources of information involved and becomes the prime mechanism for retrieving data needed to solve specific problems. The tree mode is used to communicate the commander's intent and update his estimate of the current and future situation. It is also used to maintain operational history data and to verify assessments of unit readiness and capability.

Direction of Flows

Information can flow up, down, or laterally within a ship's command hierarchy. Downward flows include directives, guidance, statements of the objective, regulations, and information of importance

or general interest. Upward flows may include organic sensor data, status or progress reports, situational information, and assessments. Lateral flows contain information that may be shared by several mission teams or tasks.

INTEGRATION PROPERTIES

The information manipulated by mission teams and systems includes both persistent and nonpersistent data. In essence, data that does not survive completion of the mission task creating it is called nonpersistent. Data integration becomes important when either type of data is shared across mission teams or tasks. There are four ways of sharing data. Direct transfer is most efficient when real-time integration is required. File-based transfer is easiest to implement. Communication-based transfer is used for open systems and distributed environments. Repository-based transfer can support a tightly coupled, consistent task environment. The goal is to maintain consistent information, regardless of how parts of it are transformed in task execution. Key data integration properties (See Reference 9) are displayed below and discussed in the following text.

Interoperability - How much work must be done for one mission team or control system to manipulate the data produced by another?
Nonredundancy - How much of the data held by one mission team or control system can be duplicated by, or derived from, the data managed by another?
Consistency - How well do different mission teams and control systems cooperate to maintain a unity of effort when sharing data?
Data Exchange - How much work must be done to make nonpersistent data generated by one mission team or control system usable by another?
Synchronization - How well does each mission team or system communicate the changes it makes in the values of shared nonpersistent data?

Interoperability

Major warfare systems increasingly are network systems that may have nodes in the air, at sea, and on land. U.S. and allied forces will have to employ common approaches to doctrine, planning, and battle space awareness data if interoperability is to be assured. Suppose that one mission team or system is using a particular view of some data in the mission control environment. Another team may need to use the same data but see it at a different level of detail or utilize another scheme for representation. Many kinds of information can be shared, including documents, voice, imagery, graphics, specifications, computer programs, computer program data, and metadata (descriptions of data structures and relationships). Whatever kinds of information are involved, it is often difficult to say precisely what is being shared. The difficulty factor arises because what is being shared depends on the kinds of agreements that exist between the components sharing data. Two teams or systems may wish to use different symbol sets or even different semantics for data from a single source. The following categories indicate the levels of agreement that can exist.

- ***Carrier.*** Agreements at this level allow data to be shared by teams or systems independent of syntax or semantics. Use of a common byte stream, as in UNIX, is an example. Without such agreement, each node's output must be converted for use by another node. But since the nodes have no shared understanding of the data, each must analyze the entire stream.

- ***Lexical.*** Agreements at this level establish a common understanding of the basic tokens of data (or symbols) being shared. Nodes are then able to interact with greater efficiency. But they still have no shared understanding of operations performed on the data. One form of lexical agreement: a list of items separated by commas.

- ***Syntactic.*** This level of agreement establishes a common syntax for shared data. Nodes may agree on a set of data structures or rules governing the set's formation. This permits nodes to avoid repeating actions to analyze, validate, and convert data structures, but there is no shared understanding of meaning and implied behavior.

- ***Semantic.*** This level of agreement establishes a shared understanding of data structures and operations performed on those structures. This can be done either by embedding knowledge of data structures and operations into systems and procedures or by creating a repository of information about them. This level of integration is an enabling factor for task automation.

- ***Method.*** This level of agreement establishes a shared understanding of the method, or context, in which the data is to be shared. Some knowledge of context is essential for nodes to understand how data elements interact. Method-level integration comes close to process integration, since it implies each node understands the overall work process to some extent.

These categories have some interdependence and the boundary lines are not crisp. The difficulty of implementing agreements increases with the scope of understanding to be shared and can become an obstacle to openness.

The role of information standards is to document the agreements set up to enable interoperability. In essence, they define a logical view of data (meaning and use context) independently from the processes that create or use it and make it possible to maintain and share that view across many processes. Both Army and Joint Technical Architecture documents (References 10 and 11) cite an extensive list of relevant standards. Both mandate use of IDEF tools to model information flows and cite the Defense Data Dictionary System (DDDS) as an authoritative source for data standards. Managed by DISA, the DDDS is a central database that includes standard data entities, data elements, and data models.

An information model is a representation at some level of detail for a set of real-world processes, products, and/or interfaces. Models of three types (for activities, data, and interfaces) are often created. Activity models describe the procedures (automated and manual) a team must perform to achieve its mission, with particular attention to required information. Data models, developed from the information requirements given by activity models, define entities and their data elements and illustrate the relationships among the entities. Interface models tie disparate activities together to represent mission processes and tasks. Interface models may be customized to fit particular teams or systems and are considered later, in the section on Infrastructure.

Nonredundancy

Redundant information in a database, whether duplicated or derived, is undesirable because it is difficult to maintain consistency. Nonredundancy is relevant even if different teams and systems store their information in the same database. In general, mission control operations can benefit from efforts to rationalize information flows. Air dominance operations, as an example, involve many different systems and target types. Ballistic and cruise missiles, fixed wing and rotary wing aircraft, and unmanned air vehicles are target categories with different tracking, identification, and engagement characteristics. Missiles, guns, electronic countermeasures, and manned aircraft are used to engage threats, while a large array of sensors provides target motion and characteristics data. Since a weapon system may be served by multiple sensors as well as multiple target or track databases, rationalizing information flows involves integration of sensing and fusion resources as well as data flows. Clearly there is much room for improvement.

Today's front line surface combatants are required to perform at sea for long periods of time. A multimedia archive system is needed to provide local repositories of tactical, technical, and administrative information for use during these periods. Today, database support tends to be stovepiped, and it is difficult to maintain compatibility across ship classes. At times, archive data can become so out of date or lacking in detail as to interfere with efficient operation. Effective integration calls for adoption of a single repository to provide paperless access to archive data on a shipwide basis. This will create many opportunities for eliminating redundant data. The way to begin is by reaching agreement on a general, extensible information model, plus an object manager to perform data storage, query handling, update, and security functions. The repository must support concurrent access to multiple versions of many objects, ranging in size from very small to very large.

Consistency and Data Management

Given the need for a shared understanding of the battle space, the Navy can't risk not having a common situation plot. A fundamental goal is to provide for a common database with the ability to deliver appropriate data to users at different locations and levels of command, regardless of the actual equipment used. This demands the ability to maintain several different views of the battle space and to propagate changes between such views, allowing mission teams and individual operators to work at different levels of responsibility and utilize different representations (symbology). Each team has its own roles and capabilities and may have different battle doctrines, rules of engagement, and

situation displays. Resulting views will differ in many respects, but basic consistency must be maintained to establish a unity of effort. Each view should be derived from a common global process model to ensure consistency in terms of view creation and change propagation. The information must be timely and it must have the ring of validity; it can't be just someone's interpretation of the battlefield.

Data Exchange and Synchronization

This area is very similar to interoperability, but applies to nonpersistent as well as persistent data. Future theater warfare operations will call for the collaboration of multiple mission teams controlling different parts of a joint operating process. Cooperative engagement was cited as an example earlier. There is also growing interest in the concept of a virtual land attack system that uses interacting air, space, and surface resources to detect, target, and attack in real time. Elements of the concept include creation of a theater planning node capable of operating at sea, on land, or in the air; an object-oriented model of adversary operational structures; and a modular family of munitions that can be allocated to generate highly effective attacks at relatively low cost. Generation of high-quality fire control loops across multiple participating units depends on the accuracy, latency, reliability, and consistency of associated data exchange and synchronization processes. This involves tremendous demands on data processing, simulation, and display resources. For example, high-capacity data storage is essential to provide fast access to photographic information or to three-dimensional terrain data. Such demands will make it necessary to adopt new computer architectures, based on parallel and distributed processing.

CONTROL INTEGRATION

The goal of control integration is to allow the flexible use of resources to accomplish mission tasks in accordance with user preferences. Teams must share functional resources to support flexibility in team structure and access to resources. Within this context, control and data integration are complementary. Data integration deals with data representation, transfer, conversion, and storage concerns, while control integration deals with control transfer and service sharing concerns. The control environment must accommodate a large array of possible operating modes and configurations to permit effective service in a wide variety of scenarios and environments. Consequently, it is necessary to support and integrate results from three main groups of processes. Mission planning processes support efforts to define what the mission teams will do. Readiness management processes provide access to essential resources. Mission control processes execute a chosen course of action.

In a sense, a backbone environment consists of the services it makes available to users. To the extent that services can be standardized across mission teams, it becomes possible to create a general task environment for mission control. Given a joint approach to doctrine, mission planning, and battle space awareness, interoperability can also be improved. As suggested by Figure 7, the general task environment (GTE) is likely to provide only part of the resources available to each mission team. An operating environment for software and hardware resources (SHOE) and a specific task environment (STE) of assets and capabilities that are unique to a mission area will complement the GTE. Ideally, a service offered to any team should be accessible to all, and the use of one service by a team should not limit appropriate use of any others. However, simply eliminating incompatibilities between component systems, so that the mission teams no longer perform tedious data entry or extraction tasks because computers won't communicate, would make a good beginning.

MISSION PLANNING

Information flows are one of the critical elements in the conduct of operating tasks and a central factor in design of a backbone environment for use by mission teams. Finding ways for mission teams to utilize massive amounts of information from external sources is especially important in mission planning. Existing systems tend to limit access to such data and make it difficult for joint and coalition forces to develop an integrated command and control structure.

Commonality in mission planning begins with a shared concept of operations and the associated command structure, and in many regional conflicts decision criteria for key phases of the campaign will have a joint flavor. Today, for example, it can be

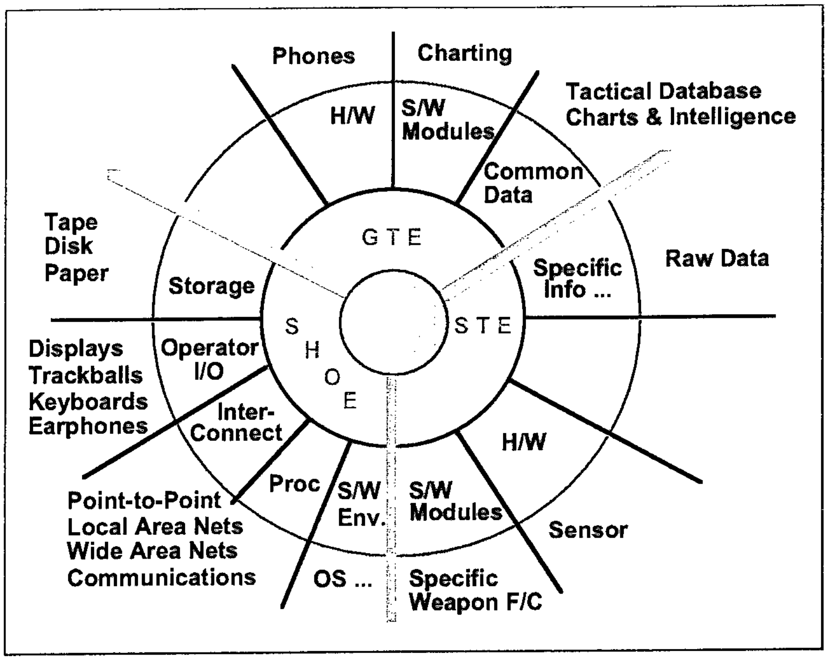


FIGURE 7. GENERIC BACKBONE ENVIRONMENT

awkward to coordinate mission planning with embarked forces because ships do not employ the rapid reaction planning process used by the Marine Corps. In another decade it can be expected that joint mission planning, using simulation-based planning methods, will enable coordination across an entire theater of war. Even a diverse mix of forces will then be able to develop and execute a shared concept of operations with confidence. Another step that can be expected is the integration of mission planning with training systems. By tailoring simulation and modeling assets to the actual operating area, and incorporating current intelligence, it will become possible to generate a visual rendering of mission plans on a synthetic battlefield. The notion of "just in time" training then becomes meaningful. Beyond training for a given course of action and on-the-fly revision of training scenarios, this means new capabilities for plan evaluation and rehearsal of critical tasks.

At present, however, mission planning is a manpower-intensive undertaking with a number of limitations. In particular, various process improvement efforts indicate that mission teams without extensive experience in a specific operating area could benefit from automated planning support. This system would assist mission teams in the following areas:

- **Data Structures.** Lack of standard data formats leads to interoperability problems between the planning systems used by different force elements. When one system does not "talk" to another, mission teams may have to manually transfer data from one digital system to another. Data translators and bridges are only a temporary fix; data elements and planning processes should be standardized as the systems go through modernization and upgrade cycles.

- **Battle Doctrine.** As operations become more complex and difficult, the role of automation becomes more important. Automated doctrine capabilities can help to cut reaction times further, enhance tactical flexibility, reduce operator stress, and tailor response sequences to prevailing conditions. Better capabilities can also be provided for visualizing the effects of alternative tactical doctrines.

- **Communications Setup.** Planning tools to assist mission teams in establishing a communications profile tailored to the operating environment should be provided. In advanced versions, the goal should be to load a predefined plan and have the communications profile automatically adjusted as necessary to implement the plan. Plans would be assessed on a continuing basis, with summary feedback to operators on connectivity and traffic statistics. The idea is to simplify the mission team's operating processes.

- **Display Management.** Similarly automated capabilities should be provided for planning and setup of operational displays. The idea is to tailor displays to what operators need to know in the actual operating environment. This involves use of information layering and visualization techniques, advanced graphical user interfaces, and ways of automating extraction of essential information.

- **Virtual Teams.** A virtual command team concept can be implemented using computer-supported cooperative work tools to conduct assessments, develop or evaluate a course of action, and address technical or support problems. This enables mission teams to work together across department or service lines.

- **Message Processing.** A broad range of information is sent to the ship by naval message traffic. This

traffic is generally maintained as separate paper message logs, making it somewhat difficult to cross-correlate with ongoing events or tactical displays. Command needs paperless delivery and access to operations messages, along with automated parsing, sorting, reformatting, and display of pertinent information. In short, provision should be made to strip the message data of unwanted formats and transform it into something usable by a human being.

- **Reference Data.** Historical data about the environment, previous operations, and contingency plans can help newly deployed mission teams learn what to expect and what has worked in the past. In addition, CD ROM "juke boxes" can be provided to supply tactical references (joint and service publications) for use in planning. Teams without extensive experience in the operating area could benefit from access to a consolidated database.

- **Data Fusion.** Mission teams continue today to plan operations using large charts and yellow adhesive notes. Enormous efforts are made in gathering data and trying to make sense of the operational situation. Access to a large-screen display and automated planning tools would be a big step forward.

READINESS MANAGEMENT

The role of readiness management is to balance available resources against tasks. Local readiness functions monitor capabilities in each major mission area and provide status information to the responsible mission team. The mission team determines when intervention is necessary and initiates the proper response. The ship-level readiness function collects technical status data from mission teams and must transform it into a description of available warfighting capability for the command team. But this arrangement is far less than ideal. First, the command team has limited connectivity to the different areas within a ship, so available data is somewhat limited. Second, reports from individual mission areas can be inconsistent, as each has different data structures. Third, even if the available reports are consistent, it may be quite difficult to transform the technical data into operational information. There is a need for a "what if" tool to help in assessment of mission impact, estimation of recovery time, and allocation of resources. Readiness can be defined and quantified adequately only in terms of an operating profile, tailored to some mission environment and tasking.

Configuration Management

The basic functions in this area include identification, control, status accounting, and verification. Distributed control systems typically demand a sophisticated level of configuration management because of the frequency of field reconfiguration and upgrade. Configuration management tools are used to define node addresses and the relationships between the nodes, to add new nodes and to replace any defective ones, and to test any node and any function needed to properly start up and maintain the system. Configuration management requirements come as a surprise to many people moving from centralized to distributed control structures. The earlier system never had to replace nodes or address them. Relationships between its nodes (sensors and actuators) were usually programmed into the system. Changes (e.g., adding sensors or actuators) were difficult and expensive to make. But the "necessary evil" of system management is really an enabling tool to harness the new capabilities of distributed control structures, including autonomy of nodes, flexibility of configuration, and ease of maintenance.

Fault Management

Total failure in warfighting control systems is unacceptable. Fault management services allow a system to react to the loss or incorrect operation of system components at various levels. Because no one wants to go to war with weapons that don't work, a general task environment for warfighting control must provide for predictable behavior under a wide range of conditions, based on well-tested systems with built-in protection against the loss or incorrect operation of system components at various levels. Customary methods of design for fault-tolerant operation include the use of ultrareliable components, redundancy, component diversity, casualty mode operation, and comprehensive capabilities for fault detection and recovery.

One of the major concerns in design of warfighting control capabilities is that an adversary may attempt to attack weak points of the control structure itself. Any flaw in weapons or battle doctrine, poor safety or security, or susceptibility to interference effects represents a possible failure mode.

Performance Management

Services in this area allow resources to be managed efficiently. System management includes capabilities for defining and managing user access, devices, file systems, administrative processes, queues, machine/platform files, authentication, authorization of resource use, and system backup. Performance aspects of hardware, software, and network components must be monitored and subsequently made available to the system manager. The manager must then have access to resources and parameters with which to tune the system to meet performance targets. In this process, the system manager must monitor various triggers for function migration such as failure or repair of hardware components, mission phase or workload change, requests, and timed events.

Data extraction, recording, playback, and reconstruction capabilities are essential aids to system performance management. To aid mission teams in conducting reviews of recent operations, capabilities should be provided to record pertinent information for at least 24 hours. The archiving feature should provide snapshot capabilities that allow an operator to select any period of archive data for permanent storage. In addition, playback capabilities should be provided that allow an operator to replay any time period in the last 24 hours to any station in the ship, without disrupting live operations. Multimedia data storage will be needed eventually.

Security Management

Achieving adequate security protection may involve various approaches, including the control of physical access to facilities, the invocation of particular security mechanisms in computing platforms, and the control and protection of information transferred between elements. Security services that need to be provided either through implementation in the information systems or through the exercise of physical facility and administrative control procedures include identification, authentication, access control, assurance of resource integrity, service standards, accounting, and auditing. Where possible, the backbone environment should reduce or eliminate the need for separate or dedicated systems to process information controlled by different security policies. This means providing an application platform with security protection adequate to permit simultaneous processing of multiple tasks, subject to multiple security policies of any complexity or type, including policies for sensitive unclassified information and for multiple categories of classified information. This will permit system use to support multiple missions with varying sensitivity and rules for protected use. It will also permit the system to give simultaneous support to multiple users with different security attributes.

Force and Warfare Systems

The importance of shipboard readiness management functions has long been apparent. A need for readiness management capabilities is emerging at force and warfare system levels as well. The problem of asset allocation in a warfare system inherits all the difficulties of ship-level readiness management. But warfare systems have little access to shipboard readiness information. To avoid creating a communications bottleneck, the data transmitted to higher command should include only summary data. To develop a basic asset management capability at this level, it is also necessary to address operating concepts and mission profiles, control structure and techniques, and associated component capabilities.

For warfare systems, readiness management includes determining the source and allocation of sensors, systems, and data assets among participating units to enable coordinated mission operations. Ability to disperse tasks among participating units is both a strength and a weakness in the face of partial loss of function. The strength is that the mission can be at least partially successful, even when one or more of the participating units is lost. The weakness is that relatively complex reconfiguration procedures across the overall warfare system may be necessary to maintain continuity after a disruption. The resultant task is developing reconfiguration strategies to retain maximum effectiveness in the event of unit loss, sensor system failures or jamming, or communications losses.

Embedded Training

While shore-based training is the genesis of a well-trained crew, on-board training provides an at-hand, versatile, and challenging method of improving crew skills while avoiding some of the costs associated with shore-based refresher training. The on-board training system must be designed and integrated in such a way as to provide training evolutions that can be tailored to the entire crew, a selected portion of the crew, or individual training. It

must allow for on-line operator manipulation of the training problem and be capable of "instant replay." The concept of stimulation is fundamental. Ownship sensors, communications equipment, data links, and weapons can be stimulated through on-board training systems. Imaginary contacts can be input to sensor systems through training systems that permit the entire ship to operate as if actual contacts were present. Through stimulation, realistic scenarios can be played out to exercise the entire ship and crew.

MISSION OPERATIONS

Mission control systems create value by mediating the execution of mission tasks under orders. A wide range of mission tasks can be performed, from maneuver and interaction with friendly units to delivery of energy against a target. This section considers the character of mission control operations. A number of associated design issues and opportunities are given in the following section.

In general, each mission team performs a series of situation monitoring, diagnosis, response initiation, and execution control activities. Discrete actions involve a sequence of discrete functions or steps (an action path) designed to achieve some desired outcome. Mission teams initiate the actions and direct their execution through a control structure that places all the machinery of a large ship at their disposal. Figure 8, which is based on a concept of integrated survivability, shows a process flow that is typical of mission control operations.

Situation Monitoring

Battle space awareness is the mainspring of mission control, and a comprehensive situation plot is vital for effective operations. It is convenient to think of situation plots as layered information structures. In such a context, the first layer might reflect the information standards, symbol sets, and screen formats used to prepare and display plot data. The second layer might consist of relatively static data taken from archives, including information about the operating area, the enemy order of battle, and similar data used to establish a context for operations. A third layer might reflect command formulation of the objectives, constraints, and plans that govern mission operations. Data received from off-board sources might make up a fourth layer, while data from organic sensors makes up another. The resulting plot contains a mix of dynamic and perishable information that is updated cyclically and provided to monitoring functions. Both human operators and computer-based systems are used to perform situation monitoring. Updates from sensors with high-volume search rates may require intensive processing. High-speed processing is then necessary to meet response time constraints. For example, future systems may have to perform threat evaluation for up to 5000 tracks within a cycle time of four seconds or less. Situation monitoring involves a number of design issues and opportunities addressed at the end of this section.

Assessment (Diagnosis)

Command decision makers need full information concerning a track's kinematic, identification, and tactical histories to support decision making that is timely and correct. Currently, this can be obtained only by concentrating on a few specific tracks over a period of time. This current process exceeds one person's span of control in a dense track environment and drives current ships to designate multiple tactical action officers (TAOs) for different warfare areas. A capability to display a concise, complete tactical history of a track is needed. This history should include kinematic history, identification indicators received and all decisions made from them, actions taken, and who performed the actions. Command personnel could then rapidly review all pertinent track information prior to making a decision.

Response Initiation

This capability is of central importance. For effective mission operations, alarms and alerts at different levels of criticality must be sensed and appropriate responses quickly set in motion. Response initiation triggers are of two types. State-based triggers are closed-loop mechanisms, produced when the system is in a particular set of states. If time is regarded as a state variable, periodic functions are initiated by this method. Operators can generally override automatic systems, generating or canceling triggers at will. Command by negation is an example. Common mechanisms should be used for message passing, event notification, and triggers across all mission teams.

Execution Control

Mission tasks are broken into a sequence of primitive steps, each implemented by a set of simple components. The components include sensors,

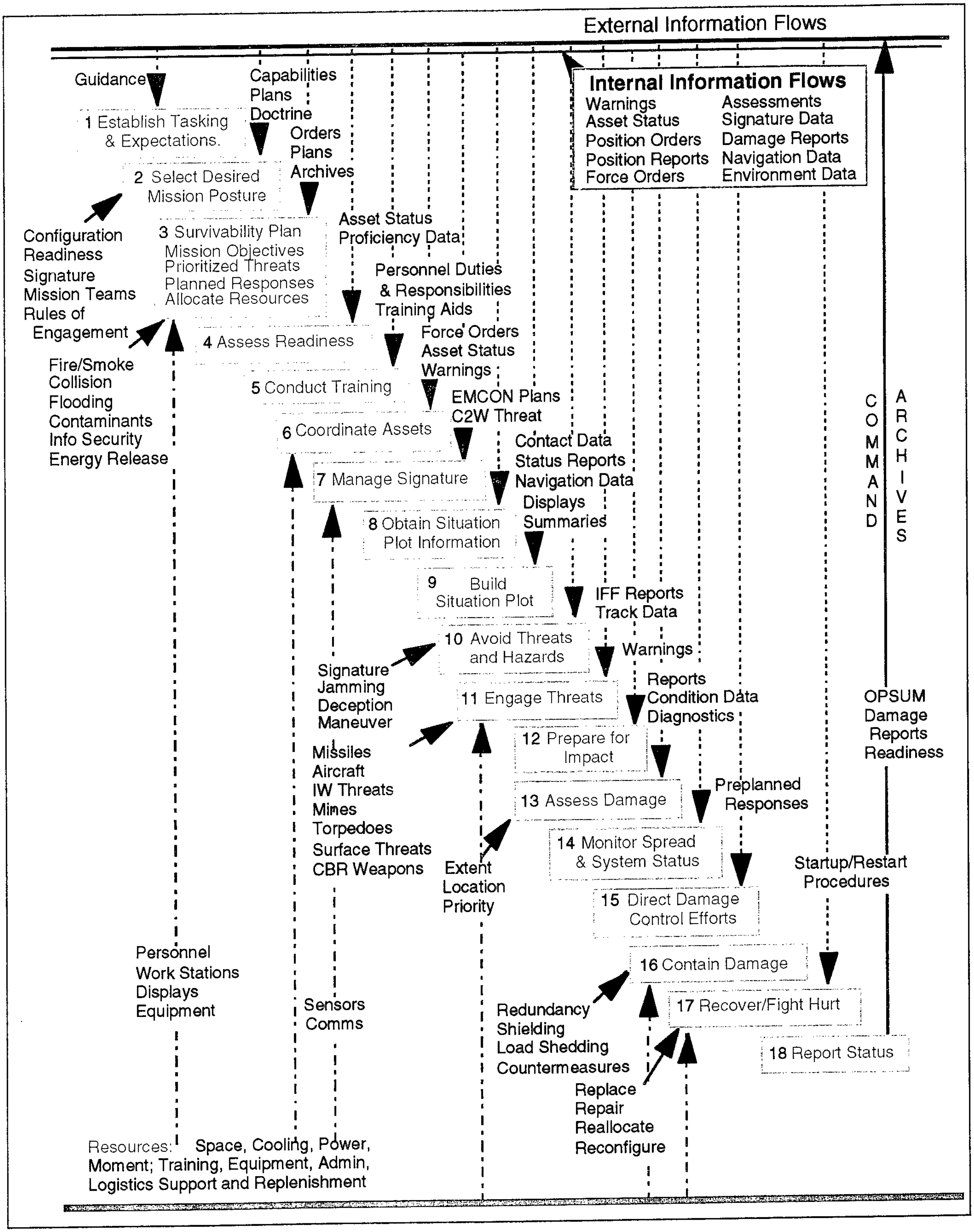


FIGURE 8. PROCESS REFERENCE MODEL - INTEGRATED SURVIVABILITY

effectors, and controllers. The sensors transform physical resources into information, while effectors transform information into actions. The control components govern resource flows. Execution of warfare tasks is controlled by a combination of manned watch stations and automated assets. In general, the task is to control target processing by on-board weapon systems. Response time (time from detection to completion of a suitable response) is a critical factor in performance and may require scheduling based on deadlines and criticality. Execution times of all tasks must be predictable and should be driven by hardware constraints rather than the time taken by human or computing elements to note alerts, select an action option, and then exercise the controls needed to generate that option. Accordingly, many elements of target processing systems are highly automated.

DESIGN ISSUES AND OPPORTUNITIES

Interoperability

Lack of a common situation plot among all components of a distributed force is a key obstacle to cooperative warfighting. A joint force commander may not even have the same plot accessible to all members of his staff, and extensive communication is then necessary to coordinate decision making. What is needed is an object database approach that offers tailored plot data (of fire control quality) for all mission teams yet is interoperable with systems used by joint and allied forces. It will take more than common data flows to accomplish such a goal. This architecture must include definition of common data interchange models, data structures, and data management services suitable for use across the many independently developed subsystems found in the mission domain.

Lack of comprehensive standards for tactical data handling is a key impediment in this area, although MIL-STD-2525A (*Common Warfighting Symbology*) may offer a convenient starting point. Another key obstacle is that computer program application entities tend to make different assumptions about what part of the software holds the main thread of control. Thus the application software used by different mission teams may complicate operational integration. As an example, consider the area of threat evaluation and weapon assignment. Many concepts for new warships contain cooperative engagement, self-defense, and area air defense systems for antiair warfare. Each component is likely to have a capability for combat identification, each may assume it holds the main thread of control, and each is unlikely to support a collaborative workstyle. What future mission teams may have to do, however, is conduct an interactive computer dialog with mission teams across a theater and manipulate plots provided by allies and coalition partners equipped with systems and application programs not shared by the U.S. Defining an object data management architecture suitable for coalition warfighting and reuse across multiple mission areas thus becomes a task of enormous scope and importance.

Multimission Displays

Since the command team supervises all mission operations, it is essential to provide display support that spans multiple mission areas. Display geolocation and range scales often vary by mission team and system. In addition, the volume of data to be handled (tracks or targets) often warrants mission-specific filtering of track data. Advanced tactical displays are needed to provide a comprehensive situation plot.

Multipurpose Sensors and Weapons

The main question considered in this effort was whether it makes sense to talk about a common system engineering framework across many different mission projects. This would mean, for the ship as a whole, the kind of flexibility and resource sharing achieved by the Vertical Launching System in handling multiple missile types or the AEGIS Weapon System in handling multiple simultaneous targets. But the adoption of a common backbone environment for control functions will create new opportunities for multipurpose functional components as well.

Figure 9 utilizes the open systems paradigm to describe the notion of a multipurpose sensing system. The diagram shows three entity types and two interface types. Application entities are viewed as report generators, turning sensor data into packages that meet user requests. One or more application platform entities provide basic sensor functions as services. The role of the resource manager is to allocate and schedule sensing resources to satisfy requests by application entities for services. Given resource availability data for external sensors, a resource manager could be utilized to request coverage from an external sensor (outsourcing), manage data formats, or establish doctrine for cooperative action to identify tracks. Users and targets are external environment entities,

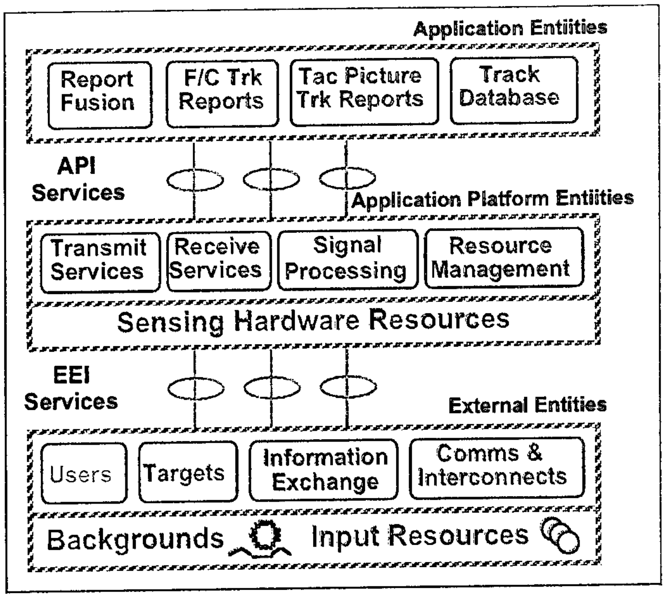


FIGURE 9. MULTIPURPOSE SENSING RESOURCES

as are sensor resource suppliers, target data providers, and sources of interference.

Historically, the creation of multifunction sensors (AN/SPY-1 Radar System) was a notable step forward in combat system automation. In this case a single, highly flexible system replaced a variety of special-purpose radars and communication links. Future systems probably ought to address the problems of situational awareness on a theater basis, working across individual ship and system boundaries. A ship designed for joint operations must not only correlate its own sensor data but also have the flexibility and architecture to use information from (or provide information to) all mission teams present in the joint mission scenario and operating environment.

Figure 10 describes another success story, the Vertical Launching System, using a similar model. In this case, missile types are the application entities. Future extension to launching of unmanned aerial vehicles (UAVs), gun projectiles, and decoys is not at all difficult to envision. These examples serve to illustrate that there is a hidden assumption underlying the way we currently design and build mission systems: information and control flows are relatively static. We may expect data in these flow paths to change as the tactical situation evolves, ordnance is expended, and new orders are received. But we do not expect or provide for continuously redefining the way system tasks are viewed. Design of sensing and engagement components for any-to-any interconnection can support a large repertory of action paths, with flexibility to create new operating modes tailored to specific operating tasks and roles. Effective use of a large repertory enhances a commander's ability to dictate the terms of action and achieve the advantages of surprise in tactical operations. Thus, future warfighting systems should be able to continuously redefine information and control flows by altering interconnection structures.

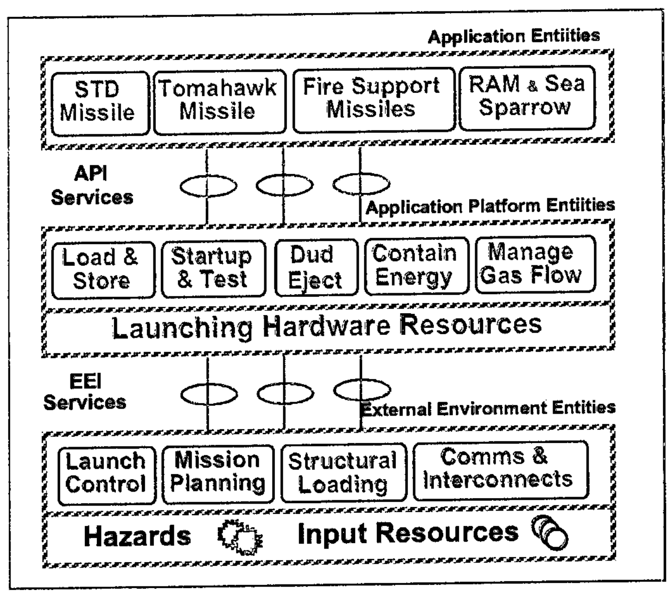


FIGURE 10. MULTIPURPOSE LAUNCH RESOURCES

PRESENTATION INTEGRATION

The goal of presentation integration is to improve user interactions with the backbone environment by reducing the associated cognitive load. This goal can be achieved by reducing the number of paradigms employed for interaction and presentation, by adopting paradigms and metaphors matched to the user's mental models, by meeting user response time expectations, and by ensuring that users get a steady flow of useful information. Key concerns are the integration of appearance and behavior, the integration of interaction paradigms, and display management.

While machines are reasonably well understood, human-computer interfaces are not. This is apparent in our designs and studies. Within the short time

allowed, studies are generally biased to quantifiable results. Improvements in hardware are easily quantified while improvements to operator interfaces must be gauged in terms of human cognitive performance. All too often, the net result is that operator interface improvements are left for another day. Since operator interfaces can account for over half of the code in any major application (see Reference 12), this is a problem. Operator interfaces not only consume application development effort, but also determine how effectively operators use the applications to collect and process information, make correct decisions, and execute required actions. If a ship is a weapon, the operator interface is its handle. Better design tools and new technologies will be used extensively in future command spaces and team work stations to create better operator interfaces. Some of the opportunities are shown in Figure 11 and discussed in the text below.

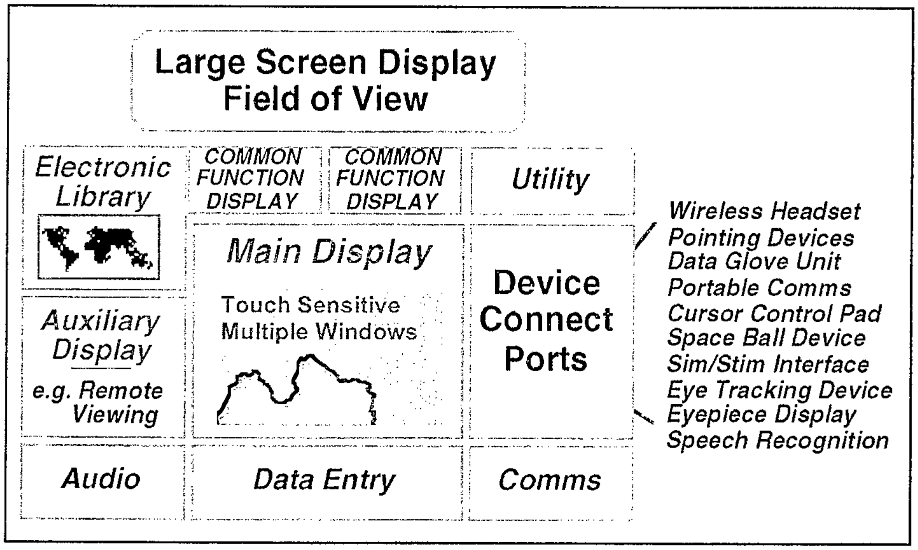


FIGURE 11. CONTROL AND DISPLAY FUNCTIONS

Communications

The services provided at individual watch stations are considered here rather than the overall communications capabilities of a ship. A key goal for development of advanced watch stations must be to automate communications management so operators can communicate effectively without diverting their attention from primary mission tasks. "Simplify" should become the watchword. External communications today are accessed by voice link to radio operators, and it is difficult for a watch stander to maintain awareness of actual connectivity and performance. Automated capabilities should include ways to establish, monitor, and maintain a target communications profile tailored to the watch position. This would be a natural extension to an automated capability for communications mission planning. Watch standers should be able to load a predefined plan and have the communications profile automatically adjusted as necessary to implement the plan. Plan assessment will be conducted on a continuing basis, with summary feedback to the operator. Information on connectivity of correspondents should also be made available. Opportunities for simplified communications also include the following:

- Current techniques for controlling voice communications depend on an assortment of switches that are physically awkward to manipulate. Future watch stations should implement these in software.
- A personal interior communication system, with features tailored to the level of command supported, can be provided. The internal communications system might then be tapped to provide a variety of services; e.g., voice link generation, video teleconference, electronic library, portable communications (handheld, wireless), and personnel location or paging (shipwide).
- Much information is passed between ships by message traffic. Future capabilities should include paperless delivery and access to operations messages, along with automated parsing, sorting, reformatting, and display of pertinent information.
- Voice recognition should be used to simplify security and authorization procedures. Automated voice can be used for prompts and acknowledgments.

Major improvements in audio equipment are also needed. Operators may be required to communicate on at least two internal nets and to monitor other nets. The noise levels in command spaces are stressful and distracting. Future headsets should be designed for comfort and provide such new

capabilities as digital audio, active noise canceling, and wireless transmit/receive.

Graphics

Future displays will use new techniques to aid tactical operators. This may include extensive use of color, windows and icons, and new display heads. The latter could include flat panel displays set into command and control consoles, large wall screens, and team watch stations using tabletop displays. Interactive screens may allow data entry by touch or through pointing devices. Volumetric (3D) displays may allow better use of the cognitive skills underlying human vision to understand the tactical situation better and quicker. They will help to increase the amount of information that an operator can interpret and act on quickly and confidently. New symbol sets will permit use of icons in tactical displays, leading to reduced training time and better retention of skills.

The idea of a virtual panel (see Reference 13) illustrates what can be done. A control panel is a common instrument for controlling devices in the physical world and also for controlling interactive computer applications. A virtual panel combines elements of both physical and computer control panels. As with the physical panel, virtual panels can be arranged to permit use of "muscle memory" in activating a control device. As with the computer panel, virtual panels are easy to design and customize, and the design can implemented by many different devices. Interfaces for a virtual panel can utilize at least three techniques: gestures, manipulation of objects, and voice command. Data gloves and touch-sensitive screens are ways of implementing the first method. The second can utilize many different device objects: button, linear slider, rotary knob, stylus, data tablet, mouse pad, keyboard, trackball, or touch-sensitive screen. The last depends on sound or speech recognition devices.

Many believe that next-generation technology must and can go beyond present concepts of virtual reality and graphical user interfaces. This involves creating a new paradigm for information visualization with metaphors suitable for a "cyberspace" environment. This will mean creating advanced systems with context knowledge of information resources to assist end users in acquiring and assembling pieces of information. Figure 12, derived from Reference 14, suggests creation of a virtual environment for information access over networks where the user can navigate freely and query globally without extensive technical knowledge of the system.

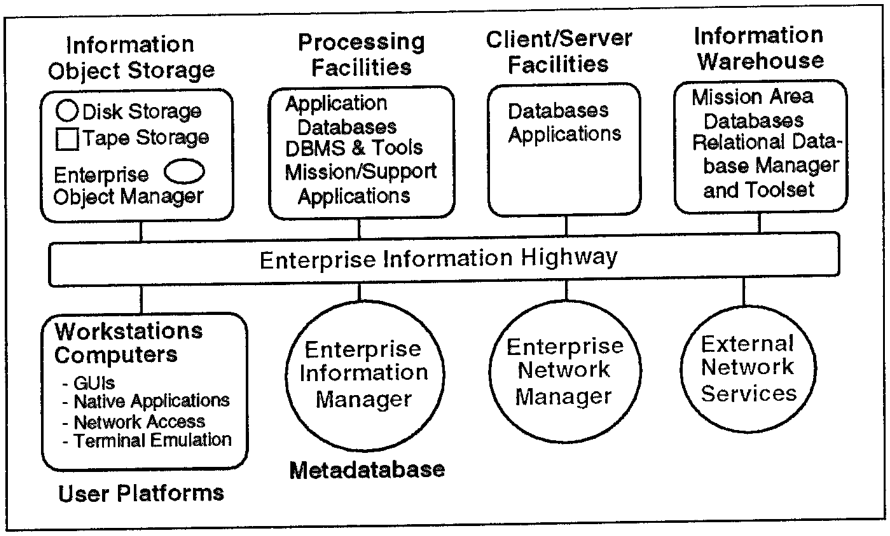


FIGURE 12. VISUAL INFORMATION UNIVERSE CONCEPT

Peripheral Devices

Future command and control consoles will likely take advantage of windowing techniques to integrate auxiliary displays into the main watch stations. However, many new peripheral devices could come into use; a number of examples appeared in Figure 11. Interactive device technology will permit the use of interactive screens and a variety of pointing techniques in future display interfaces. Touch-sensitive screens exist today and improvements in granularity and sensitivity will continue. Pointing techniques will include eye-safe lasers, photoelectric light-sensitive devices, and mouse-type devices, among others. Existing prototypes allow interaction with a display through eye contact. This has been done to let pilots get information without use of any other device.

Consoles could also provide a docking port that enables a watch stander to connect and make use of a portable computer; for example, with training or

electronic library applications. Hence future consoles may provide an array of quick connect ports to enable use of such devices.

Support Applications

Portability of application programs will be the rule in future developments. Such techniques as the following will be provided to assist operators with tactical planning and decision making tasks:

- Doctrine will be incorporated into display creation so operators or Combat Information Center (CIC) supervisors can better control the interface. This will include the use of context-sensitive help capabilities and workload balancing to maintain alertness and prevent error due to overload.

- Decision aids will be provided to answer questions and prompt users in situations fitting specified criteria. This capability should be built around a real-time database and be available for both tactical and ship service applications. For example, the system might offer cues and prompts for accessing joint doctrine, accelerated mission planning, setup of communication links, or accessing an electronic bulletin board. Arguably, there is a need to improve on today's tactical decision aids in terms of integration, standardization, and quality control across ship classes. Applications with undefined reliability and maintainability characteristics or marginal logistics support can have adverse ordnance safety implications.

- Help engines will be designed not only to speed task execution but also to help operators gain skill in the use of application programs.

- Techniques will be developed allowing operators to exercise control over the level of automation employed, the level of explanations offered for machine solutions, and the feedback provided on relevant decision parameters.

INFRASTRUCTURE

We selected shipwide computing as an example of infrastructure integration issues.* At this level we are concerned with commonalities between the backbone environment and a shipwide computing environment. A virtual machine concept holds promise for shipboard computing, but its implementation involves some critical design decisions. Perhaps the most important class of decisions is the choice of interfaces based on open system standards. First, we frame the problem in terms of a total ship architecture concept. Next, we consider applicable standards by computing function or service area. A second class of key decisions arises in formulating a strategy for isolating end use functions (so that no failure in one function can cause another to fail) while still allowing shared use of instruction processing, memory, and I/O resources. This is a final consideration.

REFERENCE MODEL

Standards have a growing role in development of multiuse software. Standard operating systems (such as POSIX), communication mechanisms (such as CORBA), programming languages (such as C), and graphical user interfaces (X-Window) have been found to make software reuse easier. Adoption of a standards-based technical architecture promotes use of standardized interfaces to achieve interoperability, portability, and scalability across all essential service areas. Known as the open system paradigm, this approach offers improved interoperability, systematic use and reuse of commercial and government products off the shelf, more competition by avoiding sole source procurement, and reduced life cycle costs.

Ideally the qualities of reliability and effectiveness needed in mission control systems can be combined with the qualities of interoperability, portability, and scalability offered by the use of an open systems approach. Figure 13 represents a ship as a layered open system containing three types of entities and two types of interfaces. The entity types are application software, application platforms, and external environment entities. The interface types include application program interfaces (APIs) and external environment interfaces (EEIs).

The application platform layer includes all system software services and all processing and display hardware. The role of an application platform is to provide services to external users at the EEI layer and to application software at the API layer. The role of the API layer is to provide application software access to the services of the application platforms.

* Topside configuration is another area that might have been chosen.

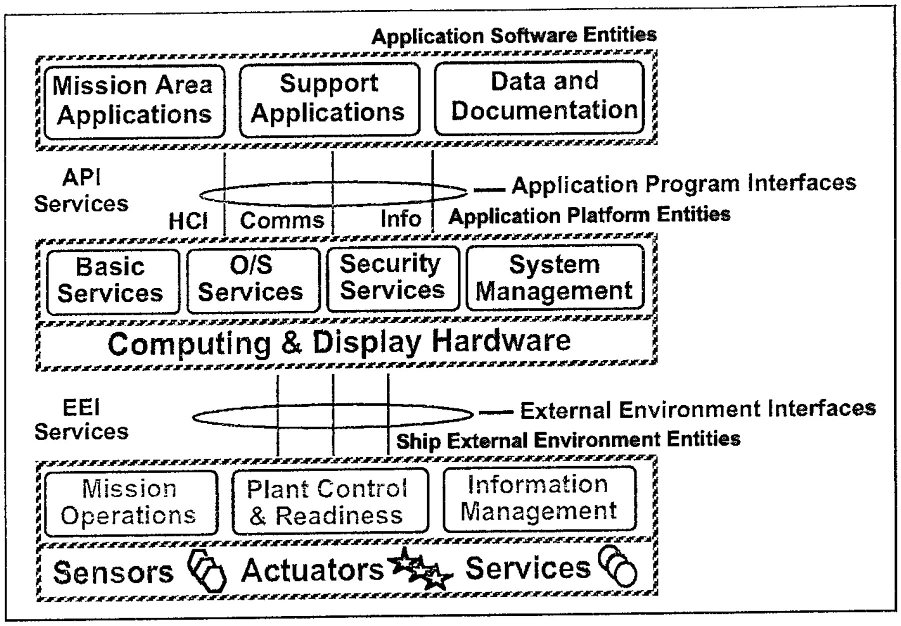


FIGURE 13. TOTAL SHIP ARCHITECTURE CONCEPT

This includes programming and operating services, communications and security services, data management and data interchange services, graphics, and user interaction services. A programmer causes an application software entity to invoke application platform services (at the API layer) by writing source code, which accesses the services when compiled and executed. As in the NIST Application Portability Profile (References 15 and 16), the basic idea is to make the software services provided at the API layer interoperable and transparent to application software. The line drawn between software services and computing hardware signifies that the hardware maker should be free to change technology and manufacturing processes while decoupling these changes from the service characteristics delivered. The two entity layers are then loosely coupled in that application platform entities become interchangeable and application software entities become portable.

The top half of the target architecture provides the logical component of mission resources; and the bottom half, the physical component. The EEI layer provides interfaces for exchange of communications, information, user interaction, and service element interaction categories. EEI and API layers are not exactly parallel because the former also provide application platforms access to ship resources such as space, power, conditioned air and water, passageways, and structural support. The external environment entities layer includes mission teams, command spaces, and ship physical resources. Control structures and people fall in the top half of the layer, while physical systems fall in the bottom half. The aim is not only to make application platform services transparent to external entities but also to make services provided by external entities transparent to application platforms. Physical services can be provided directly to end users, but increasingly such interactions are mediated by computers. Across the ship as a whole, then, any two elements providing equivalent services should be interchangeable.

APPLICABLE STANDARDS

Over the past several years many efforts have been made to establish a workable set of base standards. Some of them are listed below.

- TAFIM - Technical Architecture Framework for Information Management
- DII COE - Defense Information Infrastructure Common Operating Environment
- ATA - US Army Technical Architecture
- JTA - Joint Technical Architecture, intended to govern future C4I acquisitions
- USAF TRCs - Technical Reference Codes
- DoN Computer Resources Management
- USMC Marine Air/Ground Task Force C4I Technical Architecture

References 10, 11, and 15 through 21 provide further information. References 22 and 23 describe similar approaches generated separately for application to avionics systems.

Future surface ships are thus likely to use some set of base standards as a point of departure for definition of a shipwide computing plant. Figure 14 is based on the major service areas cited by most of the recent studies: operating system, human-computer interface, software engineering, data management, data interchange, graphics, network,

and security services. The text that follows gives a short description of each service area and associated primary standards. A brief discussion of extensions that may be necessary to establish a suitable environment for warfighting control systems follows each description.

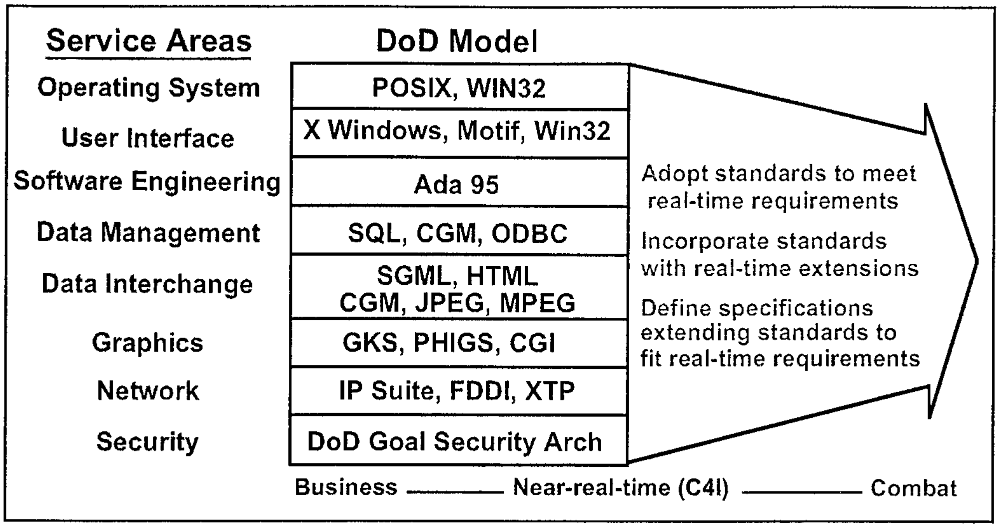


FIGURE 14. COMMON APPLICATION ENVIRONMENT

Operating System Services

This category includes the core services needed to operate and administer application platforms and to provide suitable interfaces with application software entities. Core services include kernel operations, shell and utilities, real-time monitor program, and drivers for hardware and peripherals. The JTA calls for service access through either POSIX or WIN32 APIs. Standards for distributed computing services are drawn from the OSF Distributed Computing Environment (DCE) and the work of the Object Management Group (OMG). The former supports remote procedure computing, while the latter provides common object request broker services. The DCE Authentication and Security Specification is also of interest as an emerging standard in the area of Distributed Computing Services. The X/Open Single UNIX Specification (SUS) is of interest as an emerging standard.

The importance of response time in weapon control makes the real-time monitor a vital factor in performance of warfighting control systems. The monitor's function is to ensure delivery of a required level of service in a bounded response time. Process synchronization is established in the context of given priorities, deadlines, and constraints. Fault handling and recovery are also key tasks, since any serious loss of performance can mean operational disaster. The essential fault handling capabilities include reconfiguration (where possible) and rescheduling of tasks upon processor failure. Time constraints complicate all these tasks.

Use of an operating system to control task execution in an individual computer is widely understood. The idea of orchestrating task execution across a distributed computing system is less familiar. The existence of such an overall system manager is a key factor in integrated architectures. The higher-level operating system usually maintains a hot backup, performs allocation of assets to processes, and provides fault management services. While OS kernels for real-time systems are available off the shelf, it is not a trivial matter to supply a distributed real-time executive operating system as necessary to achieve a distributed multiprocessing architecture.

Human/Computer Interfaces

This category includes the core services needed to define how users may interact with an application. The term user interface in this context is used to mean a graphical user interface (GUI). Standards are not only required for how to set up and manage graphical windows, but also for the tool kit used to generate basic display elements and so to establish the generic "look and feel" with which functions are presented to an operator.

FIPS Pub 158-1, *User Interface Component of the Application Portability Profile (X-Windows Version 11, Release 5)* is the principal base standard for this area. The X-Windows System and OSF Motif or WIN32 APIs are the major building blocks. X-Windows is a de facto standard for GUIs in open system environments. The specifications are relatively stable and any changes are expected to involve tuning rather than major changes.

A style guide is needed in order to achieve a consistent "look and feel" across different applications. The DoD HCI Style Guide (TAFIM vol. 8) provides guidelines for user interface standardization across applications developed for DoD activities and calls for each organization to detail the specific "look and feel" for its systems in addenda to the guide. Domain-specific user interface guidance developed for JMCIS and GCCS is now being incorporated in the DII COE.

Most of the mission control systems in service today have system-specific user interfaces, while the less-time-critical systems are using X-Windows and Motif. This is due partly to lack of real-time extensions to X-Windows, and to lack of style guidelines for Motif in applications that demand quick response under stress. While attention is being given to performance improvements in new versions of Motif, additional guidance will be needed to address response time and weapons safety features necessary for warfighting control applications. Some current R&D programs are making extensive use of common display kernel (CDK) and the GUILE tool kit.

Software Engineering

The procedural aspect of an application program is embodied in the programming languages used to code it. Additionally, system developers require methods and tools for development and maintenance of the applications. Core areas include programming language services, language bindings, computer-aided software engineering environments and tools, and formal software engineering methods. Ada has been a standard in the area of programming language services. However, it is useful at this point to note that the base standards apply to interfaces rather than functionality. Application programs can be written in other languages (C and C++ are frequently mentioned) as long as standard interfaces to Ada are provided to maintain interoperability.

Application programs for combat systems in service today make extensive use of CMS-2 and are the only systems that rely extensively on assembly language. Existing combat support systems use a combination of Ada and C, while C3I systems use C and other commercial languages. Little use has been made so far of object-oriented languages (C++ and Ada-95).

Data Management

Standards in this area support the storage, control, distribution, management, and allocation of persistent data and provide the means to preserve data meanings and relationships. To improve interoperability, data should be defined independently from the processes that create or use it and should be maintained and shared among many processes. Core areas of service include data dictionary and directory services, database consistency and concurrency control, distributed data services, and security services.

Entry-level SQL (structured query language) and CGM (computer graphics metafile, to store graphics) are usually cited as building blocks in this area. FIPS Pub 127-2, *Database Language for Relational Database Management Systems*, is the main standard cited for use in less-time-critical systems. ODBC 2.0 (open data base connectivity) is mandated by the JTA for both database application clients and database servers.

One of the most critical needs in mission control is to define a track data architecture that is fully interoperable with systems used by joint and allied forces and yet meets the needs of all warfare mission areas for track data of fire control quality. Such an architecture must include definition of common data interchange models, data structures, and data management services suitable for use across the spectrum of independently developed subsystems found in warfare systems. MIL-STD-2525A (*Common Warfighting Symbology*) may serve as an example and a starting point.

Today's warfare systems use system-unique solutions for track data management. Solutions are typically defined by interface design specifications (IDSs) written for each pair of interacting systems. IDS messages concerning track data are often based on similar messages in joint tactical digital information link (TADIL) standards. The less-time-critical systems make extensive use of SQL and relational database management products such as ORACLE, Informix, and Sybase. However, system-unique solutions remain customary in track data management applications.

Industry standards for database management concentrate on relational databases in a client-server environment. The standards available and under

development do not effectively address highly structured track data files. Lacking standards, individual systems have developed and continue to rely upon system-unique track file management solutions that cannot be extended into a vendor-independent distributed environment. Lack of a standard for track file data management is a key impediment to implementing distributed processing in the mission control domain.

Data Interchange

This category provides support for the interchange of nonpersistent data between applications on the same or on different platforms. Many forms of communication are used for data interchange. A list of examples might start with function calls, global variables, file transfer, object methods, shared memory, remote procedure calls, interprocess communications, and interprocessor communications. Core service areas include document, graphics, product, and electronic data interchange. Basic standards in this area include the following:

- National Imagery Transmission Format
- Standard Generalized Markup Language (SGML)
- Hyper-Text Markup Language (HTML)
- Computer Graphics Metafile (CGM)
- Joint Picture Experts Group (JPEG)
- Motion Picture Experts Group (MPEG)

The importance of highly perishable sensor data gives it a high priority in warfighting control systems, but the architecture must also deal with many other types of data including relational databases, imagery, video, and map graphics (including products derived from DMA standard products for real-time map generation). The benefits of open system design cannot be achieved if data interchange is throttled by system-unique data interchange solutions. Free interchange of data among systems developed independently requires a systemwide data architecture.

Graphics

This category provides functions required for creating and manipulating pictures. The primary type of service involved is definition and management of displays and objects. These services include definition of multidimensional graphic objects in a form that is independent of output devices, and managing hierarchical database structures containing graphics data. Basic standards include the graphical kernel system (GKS) for 2D graphics, the programmer's hierarchical interactive graphics system (PHIGS) for 3D graphics, and the computer graphics interfacing (CGI) techniques. The main additional requirement in this area is adherence to specifications established by the Defense Mapping Agency (DMA) for its mapping, charting, and geodesy products.

- The vector product format (VPF) is DMA's effort to move away from raster images of printed maps to an encoding of geographic information in a form more usable by mapping software.
- The Spatial Data Transfer Standard (SDTS) provides specifications for the organization and structure of digital spatial data transfer, definition of spatial features and attributes, and data transfer encoding.
- JPEG (Joint Photographic Experts Group) was developed to support compression of full-color still images for facsimile transmission. It is being adopted for storage and transmission of images independent of fax. Additional standards may be needed to support full motion video, video teleconferencing, and video synchronization as the associated technologies are deployed in future mission systems.

There is no standardization issue in this area, since commercial standards (PHIGS and its X-Windows extensions) are complete and widely accepted. However, the systems in service generally have vendor-unique graphics generation capabilities. Combat control systems rely on AN/UYQ-21 graphics while the Joint Maritime Command Information System (JMCIS) uses the CHART program. The issue is how to migrate to the standard PHIGS interface.

Networking

This category includes core services needed to support distributed applications requiring data access and applications interoperability in heterogeneous networked environments. Core areas of service include protocols for file access,

data communications, network management, PC support, and network security. The internet protocol (IP), transmission control protocol (TCP), and user datagram protocol (UDP) are now among the JTA base standards.

While today's mission control systems rely entirely on point-to-point interconnections, less-time-critical systems are using Ethernet running the IETF protocols. Since the Ethernet protocol does not guarantee bounded latency at any significant level of network loading, this combination does not appear suitable for time-critical warfighting control applications. Distributed information systems and distributed control systems both relocate the functionality of a complex system from a centralized processor to a collection of smaller processors. But the similarities end there. Just as a highway can be used to land aircraft, general-purpose data networks can be used to pass control messages; but the fit is not right, and problems appear when you begin to put demands on the system. Most often, general-purpose networks transmit large data files using large data packets, with low packet rates and high data rates. Distributed control systems, in contrast, must shuttle many small packets at relatively high transmission rates among a large set of nodes. The issue is not so much media volume or bandwidth as the time taken up in grinding through protocol layers to transfer messages into and out of a process. ONT's High Performance, Distributed Computing program (HiPer-D) cites an example. In a laboratory setting, several existing protocols were able to transmit only about 150 messages per second despite the use of a channel (physical link) capable of transmitting 10 Mbits per second. With the small message sizes involved, throughput was nowhere near maximum bandwidth. The protocols were not well matched to the traffic characteristics, and speeding up the channel would not resolve the problem.

Addressing is another important consideration. In a typical control structure, information can be originated by or received by any node. How packets of data are addressed to other nodes is important in determining overall system efficiency and reliability. The simplest method is broadcast addressing, in which a packet is sent to all nodes. Unicast sends the packet to one specific node, while multicast sends it to multiple nodes. Multicast is useful in warfighting control systems because the message traffic depends on external events monitored by the nodes, and one event may generate packets at many nodes. Multicast and unicast schemes also permit acknowledgments and retries and so enhance the reliability and efficiency of communications. Express transfer protocol (XTP) is cited as an applicable standard by many combat system development projects because it provides reliable multicast. Fiber-distributed data interface (FDDI) is also useful in that it offers bounded latency in reliable, end-to-end transmission of control message traffic. Commercial real-time network technology is maturing, however, and other approaches providing reliable multicast and bounded latency are expected to emerge.

Security

The TAFIM provides a blueprint for the Defense Information Infrastructure (DII), capturing the evolving vision of a common, standards-based technical infrastructure. Volume 6 of the TAFIM provides a comprehensive view of the DII from the security perspective. The DoD Goal Security Architecture (DGSA) is a generic architectural framework for developing mission-specific security architectures. While the DGSA calls for advances in security theory and technology, its concepts and principles can be incorporated into current systems.

A target for the mission control domain is to reduce or eliminate the need for separate or dedicated systems to process information controlled by different security policies. This means providing an application platform with security protection adequate to permit simultaneous processing of multiple tasks, subject to multiple security policies of any complexity or type, including policies for sensitive unclassified information and for multiple categories of classified information. This will permit system use to support multiple missions with varying sensitivity and rules for protected use. It will also permit the system to provide simultaneous support to multiple users with different security attributes. Comparable capabilities are also needed for use in a distributed environment containing a heterogeneous mix of application platforms and communication nets.

APPLICATION PROFILES

The base standards identified here and considered at greater length in DoD and service studies are not in themselves sufficient to ensure or even promote interoperability. Commercial standards typically cite a variety of non-interoperable options for many different services, to cover applications rang-

ing from cheap, uncritical consumer electronics to highly critical military and space systems. In addition, where international standards exist they are often written to include various national standards and can permit a multitude of realizations.

In DoD programs, VME tailoring has been known to create non-interoperable products.

A set of base standards typically defines an envelope that contains incompatible options for a variety of different applications. What may then be done is develop a tailored standard or profile, highlighting key areas and providing any amplifying data necessary to produce interoperable and portable implementations. This will include identification of chosen classes, subsets, options, and parameters from within the envelope defined by the base standards, as necessary to define all the services needed for a particular class of applications.

Consider a communication system profile as an example. The profile represents an operational thread through the system. It includes all the choices needed to carry out some meaningful communication-related task in an interoperable and, in some cases, portable manner. The task may be very restricted, such as an interoperable transport layer function, or very broad, such as a secure distributed file service that provides both interoperability and portability of application software entities. In the first case the profile could contain a single standard (such as TCP) required to implement a connection mode virtual circuits box. In the second case the profile could include the complete set of standards required for a secure distributed file function. A complete profile for a distributed file function would include standards for application program interfaces and distributed file functions implied in the distributed file box of the figure as well as supporting functions. The International Standard Profiles of ISO and the Standardized Profile (SP) of POSIX are examples. ISO/IEC TR 10000 (*Framework and Taxonomy of International Standardized Profiles*) and MIL Handbook 829 (*Guidelines for DoD Standardized Profiles*) are guides for defining profiles. In some cases, definition of suitable profiles could be more difficult and technically demanding than selection of the base standards.

ISOLATING END USE FUNCTIONS

While processing, memory, and I/O resources are shared in a virtual machine approach, end use functions should be isolated so that no failure in one function can cause any other function to fail. References 24 and 25 consider ways of achieving this goal through partitioning of resources in space and time. Deterministic control over the partitioning of space means that no function can prevent another from obtaining adequate memory space and that the memory space assigned to one function cannot be corrupted by the behavior of another function. This degree of control is very difficult to verify for any bus protocol that includes a destination memory address in a message. Deterministic control over time partitioning means that (a) no function's variable demand for hardware resources can prevent another function from obtaining a specified minimum level of service; and (b) it also can never disrupt the timing of access to the corresponding resources. If such control is lacking, each function can be certified only after all combinations of events in other functions, including all failure events, are considered. This adds greatly to life cycle cost.

Partitioning

One way to deal with a mixed loading of real-time and non-real-time tasks involves the use of two or more dedicated computer subsystems to separate real-time events from non-real-time events. This approach, used in several current developments, is articulated in Reference 26. Very often, supervising peripheral devices is the task that makes a host computer overloaded in real-time systems. The host may be directly connected to many data acquisition devices and system peripherals. Each communicates with the host using a handshaking protocol, whereby the host signals an external device to perform a task and the device sends an acknowledgment when the operation is completed. These peripheral devices rely on the host computer to synchronize their operation with the rest of the system. As demands increase, the host eventually becomes so burdened down that it cannot meet all deadlines.

One way of dealing with this situation is to separate real-time events from non-real-time events and use one computer for each set of tasks. Using another computer to perform all the real-time computation and throughput tasks conveniently divides the control system into two parts, an executive and a real-time subsystem. The executive consists of the host computer, I/O devices, and terminals for user interaction. Its function is to implement the operating system and to coordinate non-real-time activi-

ties. It also supervises the real-time subsystem, which includes special-purpose processors, converters, and possibly other real-time devices. Though always under control of the executive, the real-time subsystem operates independently. Asynchronous operation of these two systems is enabled by an interface placed between them. No longer burdened with heavy I/O or intensive numerical analysis, the executive computer system is freed to monitor system performance, update on-line graphical displays, record processed run data to disk, interact with the user, and perform all file manipulation functions. These tasks are termed non-real time because nothing critical happens if they are placed on hold for a few milliseconds, which is generally not true of real-time control operations. By decoupling real time from other events, the use of auxiliary processors allows closed-loop control functions to proceed using dedicated assets. The real-time subsystem does not have to contend with any slow or non-real-time events. What is probably not obvious is that a controller design with such auxiliary processors may not be any more expensive than the design having a single host computer. The reason is that the host (adjunct) computer can be less capable. This division of labor can also be realized using processors of identical make, though to a lesser extent.

Eliminating Dependencies

Many times, a commonality or dependency is created unnecessarily. Typically, integration becomes a problem when different components use different mechanisms for communication. Even a simple step such as choosing the destination of a message and embedding it into component code creates a dependency. A suitable network interface unit (NIU) can be used to eliminate dependencies. This is done by translating interface standards; performing protocol and format conversions; and absorbing differences in system data bandwidth, timing, and language. By including all interface types needed to interconnect the chosen components, an interface unit can be developed to establish the required communications.

These system integration concepts were exercised in experiments at NSWCDD, conducted with existing AEGIS and TOMAHAWK facilities. First, an Advanced TOMAHAWK Weapon Control System prototype was set up. Then a programmable network interface unit (PNIU) was connected to the central data buffer of the AEGIS Command and Decision System with a switch to permit use of either interface (CDB or PNIU) in communicating with laboratory equipments. Local area networks, work stations, and computer programs from commercial sources were installed in both facilities and a bridge was set up between the two LANs. At that point, any watch station on either network could be assigned to AEGIS or TOMAHAWK operating modes. Legacy code for tactical systems was executed in this environment *without any changes to the existing application programs.* From an architecture viewpoint these experiments, described in Reference 27, were very significant. The test equipment was introduced at the key element of a warfighting control system, the human-machine interfaces, where functionality and information content are very high. Results showed it was possible to move toward an open architecture in an evolutionary manner by off-loading low-level command and decision functions to the work stations.

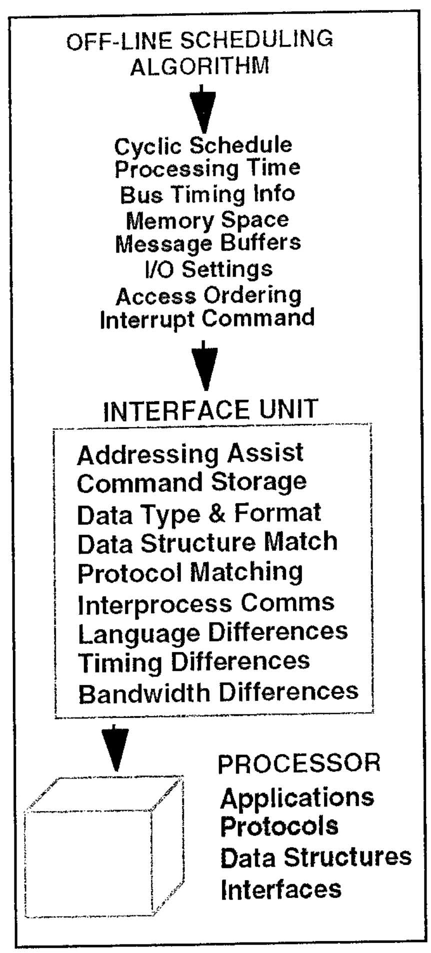


FIGURE 15. GENERAL MODEL FOR POST-FACTO INTEGRATION

Figure 15 indicates how this approach might be extended to a virtual machine environment, using a cyclic approach to process scheduling. Adoption of a cyclic- or template-based approach to scheduling permits a static allocation of processes to individual processing units, but units must be synchronized if latency control is to be assured. Scheduling methods that involve asynchronous process interruption are difficult to validate at high levels of reliability unless process interactions are restricted or exhaustive testing is performed.

In this approach, each processing node has a PNIU that contains an application-specific integrated circuit, table memory, intermodule memory, and transceiver. A sequence of commands is determined by off-line scheduling algorithm (in advance). These commands control data transfer operations such as transmit, receive, and skip; synchronization across bus interface units; and host operations (to synchronize process execution with transfer of data on the backplane). The commands are organized into cyclic loops or frames of constant length and stored in the table memory device of each NIU. Generated and stored in advance, the commands cannot be corrupted by errors in software execution or communications. Each frame begins with an interrupt command that synchronizes NIUs as they ready all modules for the next cycle of commands. Each command corresponds to a specific time window and indicates whether its NIU transmits or receives a message during the time assigned to that window. Each command points also to the intermodule memory location of the message to be passed. In this way, memory space is preallocated and contention for memory space is prevented. Memory mapping hardware in the host processor can then assure deterministic control over the partitioning of memory space. Similarly, storage of timing data in the table memories supports latency control via deterministic partitioning of time. This discussion is intended only to suggest how network interface units can be used to simplify the creation of a virtual machine design to achieve the goals of openness, isolation of end use functions, and latency control.

REFERENCES

1. NSWCDD, *Total Ship System Engineering Vision and Foundations*, TR-95/152, Dec 1995.

2. NSWCDD, *Focus on Mission Teams: Report on a Warfare System Engineering Workshop*, MP-96/83, May 1996.

3. NSWCDD, *An Overview of Total Ship System Engineering*, MP-96/87, Sep 1996.

4. D.J. Carney and A.W. Brown, "On the Necessary Conditions for the Composition of Integrated Software Engineering Environments," in *Advances in Computers*, M.V. Zelkowitz, ed., Vol. 41, Academic Press, 1995.

5. A.W. Brown and J.A. McDermid, "Learning from IPSE's Mistakes," *IEEE Software,* March 1992, pp. 23-28.

6. A.D. Churchman, W.F. Havens, C.A. Pell, S.C. Pilet, "Controls Automation and Task Allocation Program," *IEEE*, 1994.

7. Gen. John Cushman, in "Force Management Decision Requirements for Air Force Tactical Command and Control," by J. G. Wohl, *SMC* 9/81, p. 120.

8. LtCol Jack Burkett, "Tactical Information: What You See is All You Get," *Military Review*, Nov 1991, pp. 39-44.

9. I. Thomas and B.A. Nejmeh, "Definitions of Tool Integration for Environments," *IEEE Software,* Vol 9, No. 2, Mar 1992, pp. 29-35.

10. *Army Technical Architecture Version 4.0*, approved by the Army Acquisition Executive and the Vice Chief of Staff of the Army, Jan 1996.

11. *Preliminary Coordination Draft of DoD Joint Technical Architecture,* Version 0.7, Defense Information Support Agency (DISA) and C4I Integration Support Activity (CISA), attn: DISA/JEBE, Apr 1996.

12. B.A. Myers and M.B. Rosson, "Survey on User Interface Programming," *Proceedings CHI'92*, New York, ACM, pp. 195-202.

13. S.A. Su and R. Furuta, "The Virtual Panel Architecture: A 3D Gesture Framework," *IEEE Computer*, 1993, pp.387-393.

14. C. Hsu and A. Rubenstein, "Enterprise Information Management for Global Manufacturers," *Proceedings 1994 IEEE Conference on Systems, Man, and Cybernetics*, pp. 364-371.

15. Gary Fisher, *Application Portability Profile (APP), The U.S. Government's Open System Environment Pro-*

file OSE/1, Version 2.0, NIST Special Publication 500-210, Jun 1993.

16. D.R. Kuhn, "On the Effective Use of Software Standards in Systems Integration," *Proceedings of the First International Conference on Systems Integration,* IEEE Computer Society Press, 1991, pp. 455-461.

17. *DoD Technical Architecture Framework for Information Management (TAFIM),* 30 Jun 1994.

18. *Defense Information Infrastructure Common Operating Environment (DII COE), Integration and Runtime Specification (I&RTS),* Version 2.0, Oct 1995.

19. Air Force Technical Reference Codes (TRCs).

20. Assistant Secretary of the Navy for Research, Development and Acquisition, *Navy Computer Resources Management (CRM) Report,* 1995.

21. Marine Corps, *MAGTF/C4I Technical Architecture.*

22. R.B. Wray and J.R. Stovall, *Space Generic Open Avionics Architecture (SGOAA) Standard Specification,* LESC-3054-B, Dec 1993.

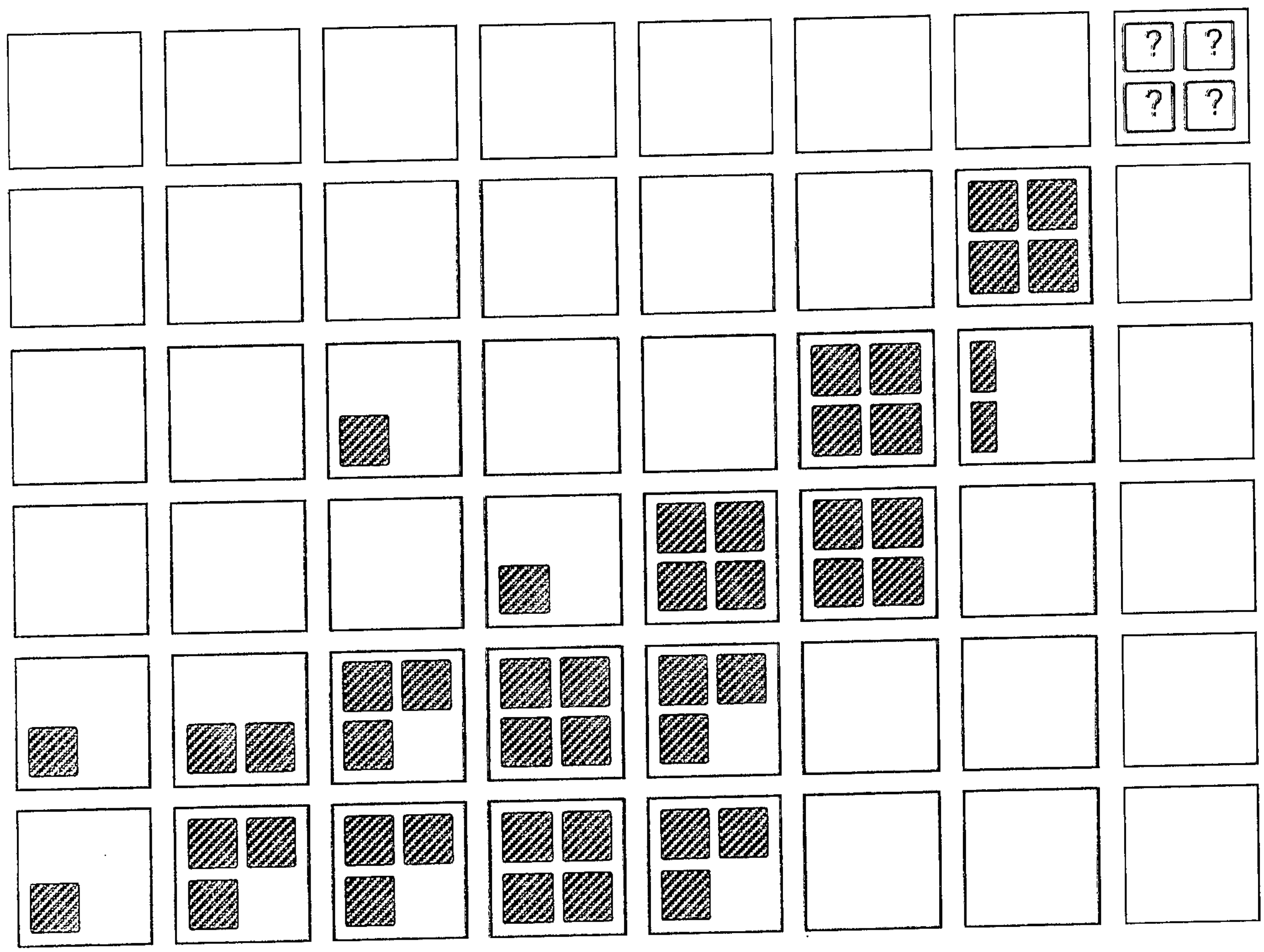
23. Naval Air Systems Command, *Advanced Avionics Architecture and Technology Review,* Final Report, 6 Aug 1993.

24. K. Driscoll and K. Hoyme, "The Airplane Information Management System: An Integrated Real Time Flight Deck Control System," *Proceedings of the Real Time Systems Symposium,* Dec 1992.

25. K. Hoyme and K. Driscoll, "SAFEbus," *IEEE Aerospace Electronics and Systems Magazine,* Mar 1993.

26. S.A. Jacklin, "Arranging Computer Architectures to Create Higher-Performance Controllers," in *Control and Dynamic Systems,* Academic Press, 1988.

27. NSWCDD, *TOMAHAWK/AEGIS Control and Display Experiments Programmable Network Interface Unit (PNIU): An Application of System Evolution Concepts,* by Terry Clayton, Systems Research and Technology Department, TR-92/487, Oct 1992.



The cover illustration is derived from a matrix used in a classic paper on computer technology to suggest the emergence of a multilayer architecture of reusable software components, akin to the multilayer architecture prevalent in computer hardware engineering. The rows of the matrix correspond to layers of functionality, while the columns represent evolving language capabilities. See Brad J. Cox, "THERE IS A SILVER BULLET;" BYTE magazine, October 1990.